For Reference

NOT TO BE TAKEN FROM THIS ROOM

Ex libris
universitatis
albertaensis

QUAECUMQUE VERA

For Reference

NOT TO BE TAKEN FROM THIS ROOM

The University of Alberta
Printing Department
Edmonton, Alberta

Digitized by the Internet Archive
in 2020 with funding from
University of Alberta Libraries

https://archive.org/details/Cabay1968

THE UNIVERSITY OF ALBERTA

DIRECT METHODS FOR LAPLACE'S EQUATION

by

© Stanley Cabay

A THESIS

SUBMITTED TO THE FACULTY OF GRADUATE STUDIES

IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE

OF MASTER OF SCIENCE

DEPARTMENT OF COMPUTING SCIENCE

EDMONTON, ALBERTA

August, 1968

THESIS
1968 (F)
27

UNIVERSITY OF ALBERTA

FACULTY OF GRADUATE STUDIES

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies for acceptance, a thesis entitled DIRECT METHODS FOR LAPLACE'S EQUATION submitted by Stanley Cabay in partial fulfilment of the requirements for the degree of Master of Science

ABSTRACT

Some finite-difference formulae to approximate Laplace's equation are derived in this thesis. An evaluation of direct methods as compared to iterative methods, for solving the systems of linear algebraic equations resulting from the finite-difference approximations, is presented for three examples of Laplace's equations.

ACKNOWLEDGEMENTS

I express my appreciation to Dr. S. Charmonman, my supervisor, for his criticism and guidance throughout the period during which the research for this thesis was carried out. I extend my thanks to Miss Susan Schultz for her time spent in typing the draft and master.

I also wish to thank the National Research Council of Canada for summer financial assistance provided under research grant NRC A-4076.

TABLE OF CONTENTS

	Page
CHAPTER I - INTRODUCTION	
1.1 Laplace's Equation	1
1.2 Purpose of the Study	4
CHAPTER II - FINITE-DIFFERENCE APPROXIMATIONS	
2.1 Notation and Basic Concepts	7
2.2 The Dirichlet Problem	9
2.2.1 Approximations of $O(h^4)$	9
2.2.2 Approximations of $O(h^6)$	18
2.2.3 Higher-Order Approximations	26
2.3 The Neumann Problem	27
2.3.1 Normal-Gradient Specified Along a Straight Line	27
2.3.2 Normal-Gradient Specified Along a Curved Boundary	33
2.4 The Third Boundary-Value Problem	40
2.4.1 The Third Problem on a Rectangle	41
2.4.2 The Third Problem with a Curved Boundary	44
CHAPTER III - TORSION OF A CROSS-SECTIONED GIRDER	
3.1 Statement of the Problem	49
3.2 Solution by Use of the Five-Point Formula	55
3.3 Solution by Use of the Nine-Point Formula	62
3.4 Numerical Results	69
3.4.1 Presentation of Results	69
3.4.2 Interpretation of Results	74

Page

CHAPTER IV - FLOW OF A PERFECT FLUID IN AN L-SHAPED CHANNEL

4.1 Statement of the Problem 78

4.2 Solution by Use of the Five-Point Formula 82

4.3 Solution by Use of the Nine-Point Formula 89

4.4 Numerical Results 95
4.4.1 Test Data and Results 95
4.4.2 Interpretation of Results 95

CHAPTER V - A FREE-BOUNDARY PROBLEM

5.1 Statement of the Problem 99
5.1.1 Physical Plane 100
5.1.2 Complex-Potential Plane 103

5.2 Solution in the Physical Plane 109
5.2.1 Method of Solution 110
5.2.2 Data and Numerical Results 113

5.3 Solution in the Complex-Potential Plane 117
5.3.1 Solution by Use of the Five-Point Formula 117
5.3.1.1 Solution of System by the Point-Gauss-Seidel Method 124
5.3.1.2 Solution of System by the Gaussian Elimination Method 125
5.3.1.3 Solution of System by Schechter's Method 125
5.3.1.4 Numerical Results 131
5.3.2 Solution by Use of the Nine-Point Formula 134
5.3.2.1 Solution of System by Point-Gauss-Seidel Method 137
5.3.2.2 Solution of System by the Gaussian Elimination Method 138
5.3.2.3 Numerical Results 138
5.3.3 Summary of Results 140

	Page
CHAPTER VI - CONCLUDING REMARKS AND SUGGESTIONS FOR FUTURE RESEARCH	142
BIBLIOGRAPHY	146
APPENDIX I - NUMERICAL DIFFERENTIATION AND INTEGRATION	
I.1 Differentiation Formulae	152
I.1.1 Formulae of $O(h^2)$	152
I.1.2 Formulae of $O(h^4)$	152
I.1.3 Formulae of $O(h^6)$	153
I.2 Integration Formulae	155
I.2.1 Formulae of $O(h^4)$	155
I.2.2 Formulae of $O(h^6)$	155
APPENDIX II - PROGRAM LISTINGS FOR THE TORSION PROBLEM	157
APPENDIX III - PROGRAM LISTINGS FOR THE CHANNEL PROBLEM	169
APPENDIX IV - PROGRAM LISTING TO SOLVE THE CANAL PROBLEM IN THE PHYSICAL PLANE	183
APPENDIX V - PROGRAM LISTINGS TO SOLVE THE CANAL PROBLEM IN THE COMPLEX-POTENTIAL PLANE	192

LIST OF TABLES

	Page
3.1	72
3.2	73
3.3	76
4.1	96
4.2	97
5.1	102
5.2	106
5.3	112
5.4	116
5.5	132
5.6	139

LIST OF FIGURES

	Page
1.1	4
2.1	12
2.2	17
2.3	20
2.4	29
2.5	34
2.6	37
2.7	40
2.8	42
2.9	45
3.1	50
3.2	54
3.3	61
3.4	61
3.5	67
3.6	68
4.1	78
4.2	81
4.3	88
4.4	88
4.5	93
4.6	94

	Page
5.1	99
5.2	100
5.3	105
5.4	109
5.5	141

CHAPTER I

INTRODUCTION

1.1 Laplace's Equation

A function $u = u(x,y)$ which satisfies the Laplace equation

$$\nabla^2 u = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0 \tag{1.1.1}$$

is called a harmonic function. Harmonic functions have the significant property of being analytic. Furthermore, the maximum principle for elliptic parital differential equations holds true, and for Laplace's equation takes the following form:

<u>Theorem 1.1</u>. *The maximum and minimum values of a function $u(x,y)$ harmonic in a region R are attained on the boundary of R.*

A proof of theorem 1.1 can be found, for example, in Sobelev [19].

The general boundary-value problem for Laplace's equation (see Forsythe and Wasow [8]) can be stated as:

<u>Problem 1.1</u>. *Let R be a simply-connected, bounded region whose boundary Γ is smooth. Let u_n*

denote the inward normal derivative of u *on* Γ *. Let* $\alpha(P)$, $\beta(P)$ *, and* $\gamma(P)$ *, piecewise sufficiently smooth, be defined on* Γ *. Find* $u = u(x,y)$ *which is harmonic on* R *and continuous on* $R \cup \Gamma$ *, subject to the boundary condition*

$$(1.1.2) \quad \alpha(P)u(P) + \beta(P)u_n(P) + \gamma(P) = 0 \ , \quad P \text{ on } \Gamma \ .$$

Due to the fact that Laplace's equation is self-adjoint [8], no term in u_s, where u_s denotes the tangential derivative of u along Γ, appears in the boundary condition (1.1.2).

Perhaps the most famous boundary-value problem in mathematics is the Dirichlet problem (the first boundary-value problem), for which $\alpha = -1$ and $\beta = 0$ in the boundary condition (1.1.2), that is,

$$(1.1.3) \qquad u(P) = \gamma(P) \ , \quad P \text{ on } \Gamma \ .$$

Some common examples of Dirichlet's problem are: the steady temperature distribution throughout a two-dimensional medium with a given temperature distribution at its boundary (Duff and Naylor [7]); the flow of a

perfect fluid in a sharply-bent, parallel-sided channel (Thom and Apelt [20]); and the steady state of a stretched membrane (Sagan [17]).

When $\alpha = 0$ and $\beta = -1$, the boundary condition (1.1.2) takes the form

$$(1.1.4) \qquad u_n(P) = \alpha(P) \;, \quad P \text{ on } \Gamma \;,$$

and the problem becomes the familiar Neumann problem (the second boundary-value problem). Two examples of the Neumann problem are: fluid flow through a sharply-bent channel (Collatz [4]); and Laplace's equation in a circle (Fox [9]). Note that if the Neumann problem has a solution, then it has an infinite number of solutions, any two of which differ only by an additive constant (see, for example, Greenspan [11]).

If α and β are different from zero for all P on Γ; that is,

$$(1.1.5) \quad \alpha(P)u(P) + \beta(P)u_n(P) = \alpha(P) \;, \; \alpha,\beta\neq 0 \;, \; P \text{ on } \Gamma \;,$$

then Problem 1.1 is called the third boundary-value problem.

Mixed boundary-value problems occur when different ones of the boundary conditions (1.1.3), (1.1.4) and (1.1.5) hold on different arcs of the boundary Γ (i.e., α or β are zero on segments of Γ but not on all of Γ). Most free-boundary problems (see, for example, Charmonman [2] or [3]) in which the position of the boundary Γ is not initially known are mixed boundary-value problems.

1.2 Purpose of the Study

For solving Problem 1.1 numerically, the region R is first superimposed by a network of meshes (which for simplicity are assumed to be square) as shown in Figure 1.1; and, the value of u is then ascertained at

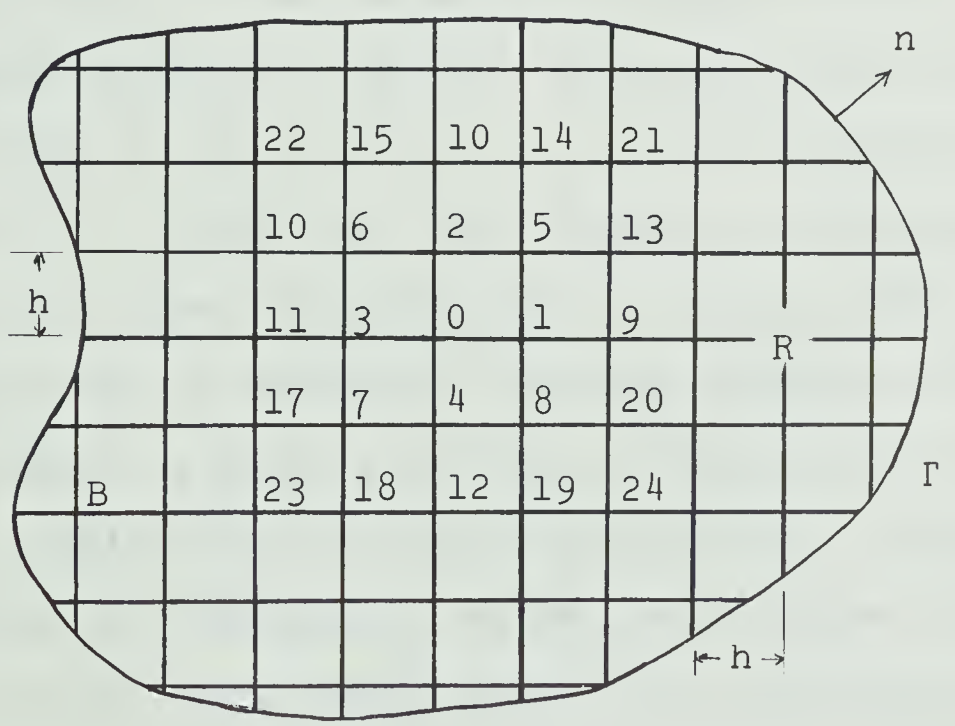

Figure 1.1 Grid System

the gridpoints in the interior of R by the method of finite-differences. In deriving finite-difference approximations, it is assumed that the function $u(x,y)$ at a grid point 0 (Figure 1.1) in the interior of R possesses a Taylor's series expansion with a sufficiently large radius of convergence. By taking linear combinations of the truncated Taylor's series of points neighbouring 0 (points 1,2,3, and 4, say) about the point 0, an expression approximating Laplace's equation at the point 0 can be found. The most difficulty will be encountered when deriving finite-difference approximations to Laplace's equation at gridpoints near the boundary (point B, Figure 1.1) especially if the boundary conditions are not of the Dirichlet type.

Problem 1.1 is therefore reduced to that of solving a system of linear algebraic equations where the unknowns correspond to the values of u at the gridpoints in the region R. Since the total number of gridpoints (and hence the number of equations) is usually quite large, of the order of hundreds, iterative methods such as the point-Jacobi and the point-Gauss-Seidel methods are most often implemented in solving the system. To speed up the rate of convergence, variations of these two methods (such as the line-Jacobi method, the line-Gauss-Seidel method, the alternating-direction methods, and the method

of successive over-relaxation) are often used. Comparative studies of these methods can be found in Forsythe and Wasow [8], in Fox [10], and in Varga [21].

The system of equations can also be solved by direct methods such as Gaussian elimination (see Ralston [16]); however, since the system is usually large, it is anticipated that rounding errors will limit the accuracy of the solution. Nevertheless, it is possible that direct methods may be sufficiently accurate and faster than iterative methods for some problems.

The purpose of this study is to investigate the use of direct methods in solving Laplace's equation. Finite-difference formulae with smaller truncation error are used to reduce the size of the linear systems. Comparisons are made between direct methods and iterative methods in terms of computer time and accuracy.

CHAPTER II

FINITE-DIFFERENCE APPROXIMATIONS

2.1 Notation and Basic Concepts

Since the functions which are harmonic in a region are analytic there, the following theorem on expansion by Taylor's series (Knopp [13]) is of consequence in deriving finite-difference formulae:

Theorem 2.1. *Let* $f(z)$ *be a function analytic in a region* R *and let* z_0 *be an interior point of* R *. Then there is always one and only one power series of the form*

$$\sum_{n=0}^{\infty} \frac{(z-z_0)^n}{n!} f^{(n)}(z_0) \tag{2.1.1}$$

which converges for a certain neighborhood of z_0 *and represents the function* $f(z)$ *in that neighborhood. The series converges at least in the largest circle about the center* z_0 *which encloses only points of* R *; and the exact radius of convergence of the series is the largest circle about* z_0 *as center in which* $f(z)$ *is everywhere defined or definable as a differentiable function.*

A convenient method of deriving formulae for approximating Laplace's equation is that due to Bickley [1]. His notation, which is also used by Thom and Apelt [20], is employed throughout this chapter.

Let $u = u(x,y)$ be harmonic in a region R with boundary Γ. Superimpose the xy-plane by a square grid-system with mesh length h as in Figure 1.1. Let the coordinates of a typical point 0 (Figure 1.1) be (x,y), so that the coordinates of the point 1, for example, are $(x+h,y)$.

If we introduce the symbolic operators

$$\xi = \frac{\partial}{\partial x} \quad \text{and} \quad \eta = \frac{\partial}{\partial y} , \tag{2.1.2}$$

then a Taylor's series' expansion of u at the points 1 and 2, for example, about the point 0 can be expressed as

$$u_1 = u(x+h,y) = e^{h\xi}u_0 , \tag{2.1.3}$$

and

(2.1.4) $$u_2 = u(x,y+h) = e^{h\eta}u_0 ,$$

respectively. Furthermore, since $\nabla^2 u = 0$, the equalities

(2.1.5) $$\xi^i \eta^j \nabla^2 u = 0 , \quad i,j=0,1,2,\ldots$$

hold true for all points in the region R .

2.2 The Dirichlet Problem

2.2.1 Approximations of $O(h^4)$ Suppose that the circle of radius h and center 0 (Figure 1.1) is contained in the region R . Then it follows from Theorem 2.1 that the circle of convergence of the Taylor's series' expansion of u about the point 0 contains the points 1,2,3 and 4. The equation

$$\sum_{i=1}^{4} u_i = (e^{h\xi} + e^{h\eta} + e^{-h\xi} + e^{-h\eta})u_0$$

(2.2.1) $$= 2(\cosh h\xi + \cosh h\eta)u_0$$

$$= \{4 + h^2(\xi^2+\eta^2) + \frac{2h^4}{4!}(\xi^4+\eta^4) + \ldots\}u_0$$

is therefore a valid one. For Laplace's equation, however, (2.1.5) holds true, so that equation (2.2.1) reduces to

$$(2.2.2) \qquad \sum_{i=1}^{4} u_i = (4 - \frac{h^4}{6}\xi^2\eta^2 + \ldots)u_0 \ .$$

We have thus proved

<u>Theorem 2.2</u>. *Let* $u = u(x,y)$ *be harmonic in* R . *If the circle of radius* h *and center 0 (Figure 1.1) is contained in the region* R , *then*

$$(2.2.3) \qquad 4u_0 = u_1 + u_2 + u_3 + u_4 + O(h^4) \ .$$

Equation (2.2.3) is commonly called the five-point formula.

If the boundary Γ for the Dirichlet problem is a rectangle, the five-point formula can be used to approximate Laplace's equation at all interior gridpoints of R . If the boundary Γ is a curve, however, different formulae must be used at points near the boundary (e.g., point B , Figure 1.1). Some irregular-star formulae to approximate Laplace's equation at such points are given in Fox [10], for example, but these have a truncation error

of $O(h^3)$. Some irregular-star formulae of $O(h^4)$ are now derived.

Let h be sufficiently small so that at least two of the points adjacent to B (points 1,2,3 and 4 are adjacent to 0 in Figure 1.1) are contained in R . Then for the typical case shown in Figure 2.1, where $\lambda_1 h$ and $\lambda_4 h$, $0 < \lambda_1, \lambda_4 \leq 1$, denote the mesh distances from points 1 and 4, respectively, we have

<u>Theorem 2.3</u>. *Let $u = u(x,y)$ be harmonic in the region R with curved boundary Γ . If u has a Taylor's series' expansion about the point B (Figure 2.1) with circles of convergence C of radius at least $2h$, then*

$$\begin{aligned}
&\{\lambda_1(3-\lambda_4) + \lambda_4(3-\lambda_1)\}u_B \\
&\quad = \lambda_1\lambda_4\{\frac{\lambda_4-1}{\lambda_4+2}\,u_5 + \frac{\lambda_1-1}{\lambda_1+2}\,u_6 \\
&\quad + 2[\frac{2-\lambda_4}{1+\lambda_4}\,u_2 + \frac{2-\lambda_1}{1+\lambda_1}\,u_3] \\
&\quad + 6[\frac{u_4}{\lambda_4(\lambda_4+1)(\lambda_4+2)} + \frac{u_1}{\lambda_1(\lambda_1+1)(\lambda_1+2)}]\} \\
&\quad + O(h^4) .
\end{aligned} \tag{2.2.4}$$

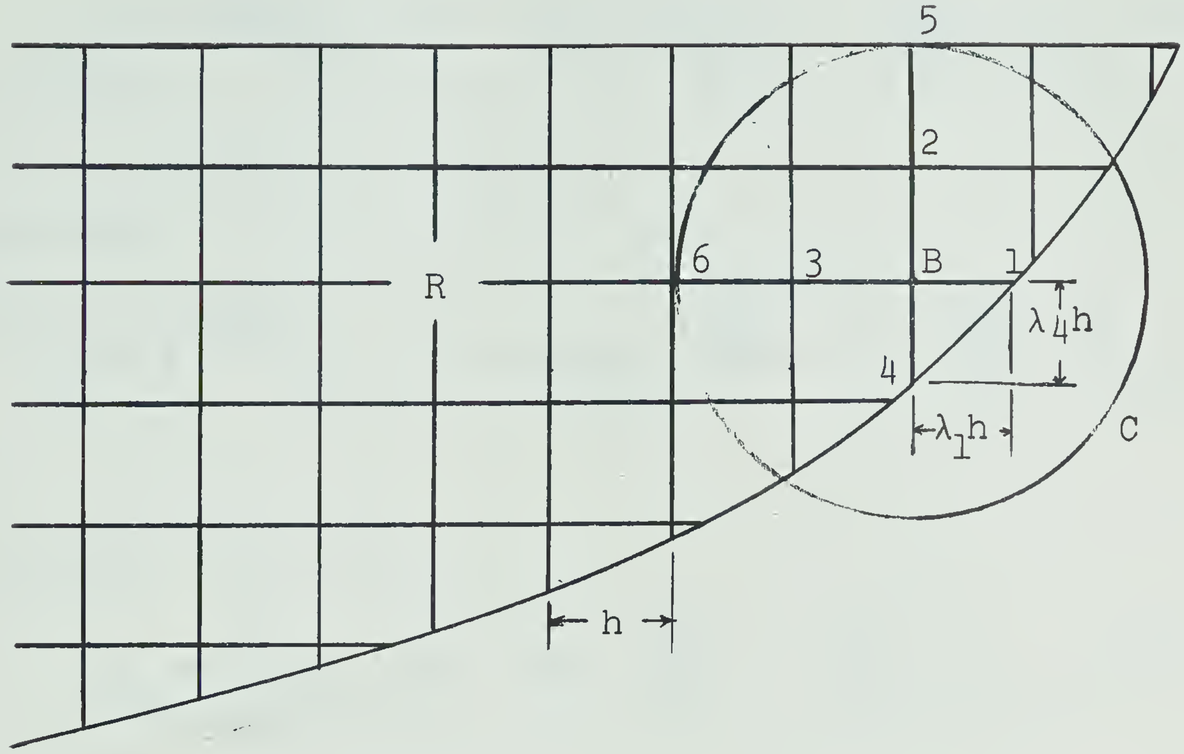

Figure 2.1 Irregular-Star of $O(h^4)$ - Positive Slope

Proof: Expanding u_1, u_3, u_6, u_4, u_2 and u_5 by Taylor's series about the point B , we obtain

(2.2.5)

$$u_1 = e^{\lambda_1 h\xi} u_B = \{1 + \lambda_1 h\xi + \frac{\lambda_1^2 h^2}{2!} \xi^2 + \frac{\lambda_1^3 h^3}{3!} \xi^3\} u_B + O(h^4) ,$$

(2.2.6)

$$u_3 = e^{-h\xi} u_B = \{1 - h\xi + \frac{h^2}{2!} \xi^2 - \frac{h^3}{3!} \xi^3\} u_B + O(h^4) ,$$

(2.2.7)

$$u_6 = e^{-2h\xi}u_B = \{1 - 2h\xi + \frac{4h^2}{2!}\xi^2 - \frac{8h^3}{3!}\xi^3\}u_B + O(h^4) ,$$

(2.2.8)

$$u_4 = e^{-\lambda_4 h\eta}u_B = \{1 - \lambda_4 h\eta - \frac{\lambda_4^2 h^2}{2!}\xi^2 - \frac{\lambda_4^3 h^3}{3!}\eta^3\}u_B + O(h^4) ,$$

(2.2.9)

$$u_2 = e^{h\eta}u_B = \{1 + h\eta - \frac{h^2}{2!}\xi^2 + \frac{h^3}{3!}\eta^3\}u_B + O(h^4) ,$$

and

(2.2.10)

$$u_5 = e^{2h\eta}u_B = \{1 + 2h\eta - \frac{4h^2}{2!}\xi^2 + \frac{8h^3}{3!}\eta^3\}u_B + O(h^4) .$$

The equality

$$\eta^2 = -\xi^2$$

was used in equations (2.2.8), (2.2.9) and (2.2.10). We wish to obtain a linear combination of the above six equations (2.2.5) - (2.2.10) which is independent of the

terms containing ξ, η, ξ^2, ξ^3 and η^3 .

Multiplying equations (2.2.5) - (2.2.10) by the constants c_1, c_3, c_6, c_4, c_2 and c_5, respectively; adding the resulting equations; and setting the coefficients of the terms containing ξ, η, ξ^2, η^2 and η^3 to zero, that is,

$$(2.2.11) \qquad \begin{bmatrix} \lambda_1 & -1 & -2 & & & \\ & & & -\lambda_2 & 1 & 2 \\ \lambda_1^2 & 1 & 4 & -\lambda_2^2 & -1 & -4 \\ \lambda_1^3 & -1 & -8 & & & \\ & & & -\lambda_2^3 & 1 & 8 \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \\ c_2 \\ c_5 \end{bmatrix} = 0 \; ;$$

we obtain

$$(2.2.12) \qquad \sum_{i=1}^{6} c_i u_i = u_B \sum_{i=1}^{6} c_i + O(h^4) \; .$$

The system of equations (2.2.11) has one degree of freedom,

so that infinitely-many solutions exist. One such solution is

$$c_1 = 6\lambda_4(\lambda_4+1)(\lambda_4+2)$$

$$c_4 = 6\lambda_1(\lambda_1+1)(\lambda_1+2)$$

$$c_3 = -2\lambda_1(\lambda_1-2)(\lambda_1+2)\lambda_4(\lambda_4+1)(\lambda_4+2)$$

(2.2.13)

$$c_2 = -2\lambda_4(\lambda_4-2)(\lambda_4+2)\lambda_1(\lambda_1+1)(\lambda_1+2)$$

$$c_6 = \lambda_1(\lambda_1-1)(\lambda_1+1)\lambda_4(\lambda_4+1)(\lambda_4+2)$$

$$c_5 = \lambda_4(\lambda_4-1)(\lambda_4+1)\lambda_1\ (\lambda_1+1)(\lambda_1+2)\ .$$

Substituting (2.2.13) into (2.2.12) and dividing the resulting equation by $\lambda_1(\lambda_1+1)(\lambda_1+2)\lambda_2(\lambda_2+1)(\lambda_2+2)$, we obtain the irregular-star formula (2.2.4).

Q.E.D.

When the point 4 is the only point adjacent to B and exterior to R (i.e., $\lambda_1 = 1$ in Figure 2.1), then

the irregular-star formula (2.2.4) yields

$$(\lambda_4+3)u_B = \lambda_4\{ \frac{\lambda_4-1}{\lambda_4+2} u_5 + \frac{2(2-\lambda_4)}{1+\lambda_4} u_2$$

$$+ u_1 + u_3\} + \frac{6u_4}{(\lambda_4+1)(\lambda_4+2)} \tag{2.2.14}$$

$$+ O(h^4) .$$

Similarly, for $\lambda_4 = 1$ in Figure 2.1, we have

$$(\lambda_1+3)u_B = \lambda_1\{ \frac{\lambda_1-1}{\lambda_1+2} u_6 + \frac{2(2-\lambda_1)}{1+\lambda_1} u_3$$

$$+ u_4 + u_2\} + \frac{6u_1}{(\lambda_1+1)(\lambda_1+2)} \tag{2.2.15}$$

$$+ O(h^4) .$$

Note that for $\lambda_1 = \lambda_4 = 1$, formula (2.2.4) reduces to the five-point formula (2.2.3).

By proper labelling of the gridpoints, the irregular-star formula (2.2.4) can be used to approximate Laplace's equation at all interior gridpoints of R . For example, if the boundary Γ has a negative slope and the region R is below Γ , then the points should be labelled as in Figure 2.2.

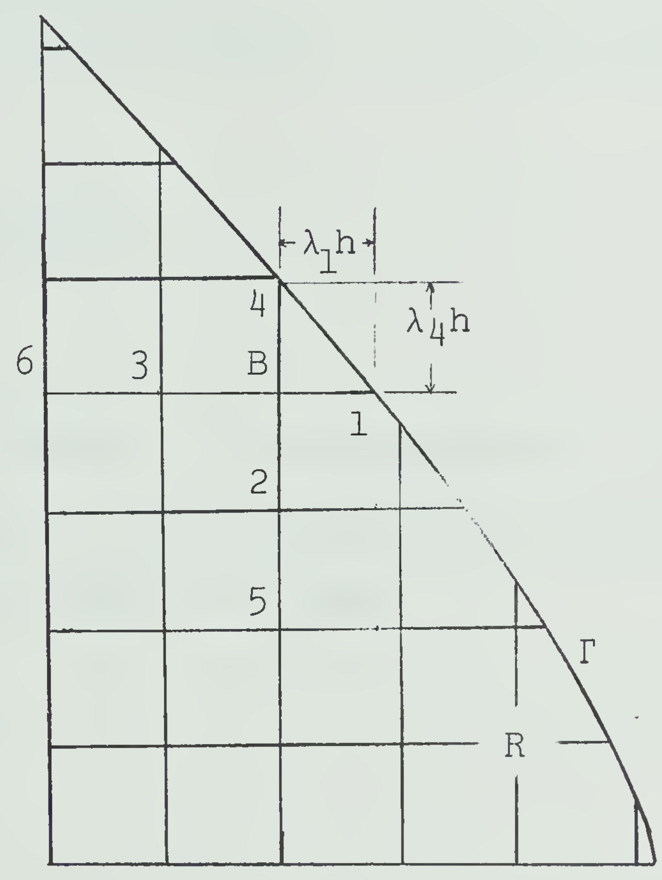

Figure 2.2 Irregular-Star of $O(h^4)$ - Negative Slope

2.2.2 Approximations of $O(h^6)$

Theorem 2.4. *Let $u = u(x,y)$ be harmonic in a region R with boundary Γ. Let the circle C with centre 0 and radius $h\sqrt{2}$ (Figure 1.1) be contained in R. Then*

$$20u_0 = 4(u_1+u_2+u_3+u_4) + u_5+u_6+u_7+u_8 \qquad (2.2.16)$$

$$+ O(h^6) .$$

Proof: Since C is contained in R, by Theorem 2.1, $u_1, u_2, \ldots, u_8$ have Taylor's series' expansions about the point 0. Thus

$$4\sum_{i=1}^{4} u_i + \sum_{i=5}^{8} u_i$$

$$= 4\{e^{h\xi} + e^{h\eta} + e^{-h\xi} + e^{-h\eta}\}u_0$$

$$+ \{e^{h(\xi+\eta)} + e^{h(-\xi+\eta)} + e^{-h(\xi+\eta)} + e^{h(\xi-\eta)}\}u_0$$

$$= 8\{\cosh h\xi + \cosh h\eta\}u_0$$

(2.2.17)

$$+ 4\{\cosh h\xi \cdot \cosh h\eta\}u_0$$

$$= 4\{4 + h^2\nabla^2 + \frac{2h^4}{4!}(\nabla^4-2\xi^2\eta^2) + \frac{2h^6}{6!}(\xi^6+\eta^6) +\ldots\}u_0$$

$$+ \{4 + 2h^2\nabla^2 + \frac{4h^4}{4!}(\nabla^4+4\xi^2\eta^2) + \frac{4h^6}{6!}(\xi^6+15\xi^4\eta^2$$

$$+15\xi^2\eta^4+\eta^6) +\ldots\}u_0 \ .$$

On substituting equations (2.1.5) in (2.2.17), we obtain the nine-point formula (2.2.16).

Q.E.D.

The nine-point formula can be used to approximate Laplace's equation for all interior gridpoints when the boundary Γ is a rectangle and for points not near the boundary when the boundary Γ is curved. For points near a curved boundary, irregular-star formulae of $O(h^6)$ can be derived.

Let h be sufficiently small so that the points 2, 3, 6, 10, 11, 8 and 9 in Figure 2.3 are all in the interior of R . If $\lambda_1 h$, $\lambda_4 h$, $\lambda_5 h$ and $\lambda_7 h$ denote the distances from point B to the points 1, 4, 5 and 7, respectively, then one irregular-star formula of $O(h^6)$ is given in the following theorem:

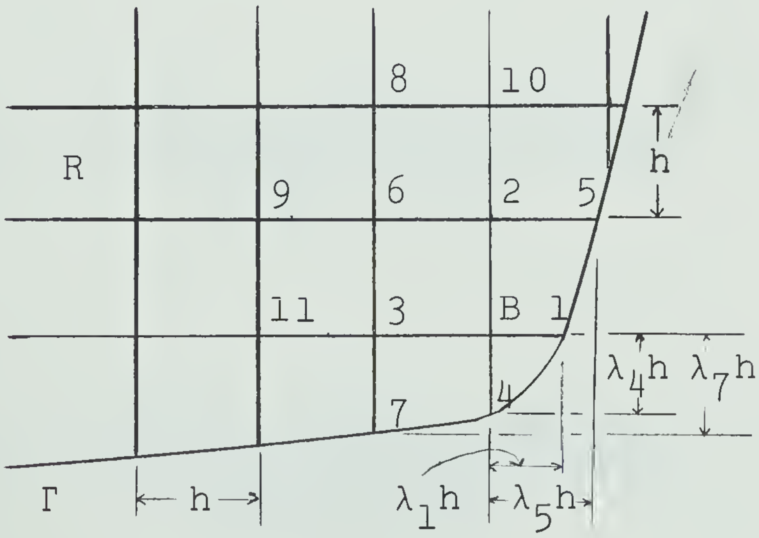

Figure 2.3 Irregular-Star of $O(h^6)$

Theorem 2.5 *Let $u = u(x,y)$ be harmonic in the region R with curved boundary Γ. If u has a Taylor's series' expansion about the point B (Figure 2.3) with circle of convergence of radius at least $h\sqrt{5}$, then*

$$u_B \sum_{i=1}^{11} c_i = \sum_{i=1}^{11} c_i u_i + O(h^6) \; ; \tag{2.2.18}$$

where

$$c_1 = 12\lambda_4(\lambda_4+1)(\lambda_4+2) \, ,$$

$$c_4 = 12\lambda_1(\lambda_1+1)(\lambda_1+2) \, ,$$

(2.2.19)

$$c_5 = \frac{-12(15\lambda_4+3\lambda_1^2+6\lambda_1-14)c_1c_4}{(3\lambda_5^2+6\lambda_5+1)k_5} \, ,$$

$$c_7 = \frac{-12(15\lambda_1+3\lambda_4^2+6\lambda_4-14)c_1c_4}{(3\lambda_7^2+6\lambda_7+1)k_7} \, ,$$

$$c_8 = -k_{14}c_1c_4 - 3\lambda_5^2(\lambda_5+1)^2c_5 - (3\lambda_7-1)k_7c_7 \,,$$

$$c_9 = -k_{14}c_1c_4 - 3\lambda_7^2(\lambda_7+1)^2c_7 - (3\lambda_5-1)k_5c_5 \,,$$

$$c_{10} = 2\lambda_4(\lambda_4+1)(\lambda_4-1)c_4 + 2k_5c_5 \,,$$

$$c_{11} = 2\lambda_1(\lambda_1+1)(\lambda_1-1)c_1 + 2k_7c_7 \,,$$

(2.2.19)

$$c_2 = -k_{14}c_1c_4 - 4\lambda_4(\lambda_4+2)(\lambda_4-2)c_4 - (3\lambda_7-1)k_7c_7$$

$$-(\lambda_5+1)(\lambda_5+2)(3\lambda_5^2+\lambda_5+6)c_5 \,,$$

$$c_3 = -k_{14}c_1c_4 - 4\lambda_1(\lambda_1+2)(\lambda_1-2)c_1 - (3\lambda_5-1)k_5c_5$$

$$-(\lambda_7+1)(\lambda_7+2)(3\lambda_7^2+\lambda_7+6)c_7 \,,$$

$$c_6 = 4k_{14}c_1c_4 + 4\lambda_5(\lambda_5+2)(3\lambda_5^2+\lambda_5+1)c_5$$

$$+4\lambda_7(\lambda_7+2)(3\lambda_7^2+\lambda_7+1)c_7 \,;$$

with

$$k_5 = \lambda_5(\lambda_5+1)(\lambda_5+2) ,$$

(2.2.20) $$k_7 = \lambda_7(\lambda_7+1)(\lambda_7+2) ,$$

$$k_{14} = 3\lambda_1 + 3\lambda_4 - 1 .$$

Proof: Since the proof of Theorem 2.5 is similar to the proof of Theorem 2.3, some of the details are omitted.

Expansion of $u_1, u_2, \ldots, u_{11}$ in Figure 2.3 by Taylor's series about the point B, and truncation after the fifth-order terms, yields eleven equations similar to equations (2.2.5) - (2.2.10) of Theorem 2.3. Some of the terms of order less than six are eliminated by making use of the equalities (2.1.5); that is,

(2.2.21)
$$\xi^i \eta^{j+2} = -\xi^{i+2}\eta^j , \quad \begin{array}{ll} j=0 & i=0,1,2,3 \\ j=1 & i=0,1,2 , \end{array}$$

$$\xi^i \eta^{j+4} = \xi^{i+4}\eta^j , \quad \begin{array}{ll} j=0 & i=0,1 \\ j=1 & i=0 . \end{array}$$

On multiplying $u_1, u_2, \ldots, u_{11}$ and their expansions by c_1, c_2, c_{11}, respectively, and adding the eleven equations together, we obtain, on setting the coefficients corresponding to terms of order less than six to zero, the irregular-star formula (2.2.18).

Forcing the coefficients of the terms of order less than six to be zero is equivalent to solving the system of equations (2.2.22), where rows 1,2,...,10 correspond to the terms containing ξ, ξ^2, ξ^3, ξ^4, ξ^5, η, η^3, η^5, $\xi\eta$ and $\xi^3\eta$, respectively. As in (2.2.11), the system of equations (2.2.22) has one degree of freedom and thus infinitely-many solutions. One solution is given by (2.2.19).

Q.E.D.

The irregular-star formula (2.2.18) of $O(h^6)$ can be used to approximate Laplace's equation at all interior gridpoints of a region. Furthermore, the coefficients (2.2.19) have the property of symmetry as do the coefficients of the irregular-star formula (2.2.4) of $O(h^4)$. However, formula (2.2.18) does not reduce to the nine-point formula (2.2.16) for $\lambda_1 = \lambda_4 = \lambda_5 = \lambda_7 = 1$. Another disadvantage of the irregular-star formula (2.2.18) is that it requires too many multiplications, at least seventy, at each gridpoint. Therefore, computations might be faster and more accurate if the interval size h is decreased near a curved boundary and lower order formulae, of $O(h^3)$ or $O(h^4)$, applied.

(2.2.22)

$$\begin{bmatrix}
-1 & -2 & & & -1 & -1 & -2 & \lambda_1 & & \lambda_5 & -1 \\
1 & 4 & -1 & -4 & & -3 & 3 & \lambda_1^2 & -\lambda_4^2 & \lambda_5^2-1 & 1-\lambda_7^2 \\
-1 & -8 & & & 2 & 11 & -2 & \lambda_1^3 & & \lambda_5^3-3\lambda_5 & 3\lambda_7^2-1 \\
1 & 16 & 1 & 16 & -4 & -7 & -7 & \lambda_1^4 & \lambda_4^4 & \lambda_5^4-6\lambda_5^2+1 & \lambda_7^4-6\lambda_7^2+1 \\
-1 & -32 & & & 4 & -41 & 38 & \lambda_1^5 & & \lambda_5^5-10\lambda_5^3+5\lambda_5 & -1+10\lambda_7^2-5\lambda_7^4 \\
 & & 1 & 2 & 1 & 2 & 1 & & -\lambda_4 & 1 & -\lambda_7 \\
 & & 1 & 8 & -2 & 2 & -11 & & -\lambda_4^3 & 1-3\lambda_5^2 & 3\lambda_7-\lambda_7^3 \\
 & & 1 & 32 & -4 & -38 & 41 & & -\lambda_4^5 & 1-10\lambda_5^2+5\lambda_5^4 & -\lambda_7^5+10\lambda_7^3-5\lambda_7 \\
 & & & & 1 & 2 & 2 & & & -\lambda_5 & -\lambda_7 \\
 & & & & & 24 & -24 & & & 4\lambda_5^3-4\lambda_5 & 4\lambda_7-4\lambda_7^3
\end{bmatrix}
\begin{bmatrix} c_3 \\ c_{11} \\ c_2 \\ c_{10} \\ c_6 \\ c_{15} \\ c_{16} \\ c_1 \\ c_4 \\ c_5 \\ c_7 \end{bmatrix} = 0 \;;$$

2.2.3 <u>Higher-Order Approximations</u> Higher-order formulae to approximate Laplace's equation, $\nabla^2 u = 0$, can be derived by the same method as were the five-point and the nine-point formulae. One such formula of $O(h^{12})$ taken from Thom and Apelt [20] is

$$7548u_0 = 780 \sum_{i=9}^{12} u_i + 512 \sum_{i=13}^{20} u_i + 83 \sum_{i=21}^{24} u_i \qquad (2.2.23)$$

$$+ O(h^{12}) ,$$

where the points 0,9,10,...,24 are labelled as in Figure 1.1.

Higher-order formulae have an advantage over lower-order formulae, of $O(h^4)$ of $O(h^6)$, in that a larger interval-size h, and thus a fewer number of gridpoints, can be used to achieve the same accuracy. This advantage is especially significant when solving Laplace's equation in a large region by direct methods on a computer, where memory storage is limited.

However, since there is a large spread of points in higher-order formulae (see Figure 1.1), different formulae must be used at points near the boundary, even if the boundary is a rectangle. Special formulae of the same order can be derived for these points, but these are no

longer symmetric about the point 0, so that the coefficients become cumbersome and the formulae inconvenient to use. An alternative is to decrease the interval-size near the boundary and use lower-order formulae.

2.3 The Neumann Problem

For the Neumann problem, formulae (2.2.3), (2.2.16), or (2.2.23) can be used to approximate Laplace's equation for points not near the boundary. For points near the boundary, different formulae which incorporate the Neumann boundary condition (1.1.4) must be derived.

2.3.1 Normal-Gradient Specified Along a Straight Line

Theorem 2.6. *Let* $u = u(x,y)$ *be harmonic in a region* R *with boundary* Γ . *Let the lower part of the boundary* Γ *be a straight line* L *parallel to the x-axis on which* u *is harmonic. Assume that* u *has a Taylor's series' expansion about a boundary point 0 (Figure 2.4) with radius of convergence of at least* h , *and that,* $\frac{\partial u}{\partial y} = \gamma(x)$ *on* L . *Then an approximation to Laplace's equation at the point 0 is given by*

$$(2.3.1)\quad 4u_0 = u_1 + 2u_2 + u_3 - 2h\gamma(x) + \frac{h^3}{3}\gamma''(x) + O(h^4) .$$

Proof: Expanding u_1, u_2, u_3 by Taylor's series' about the point 0, we obtain

$$u_1 + u_3 = (e^{\xi} + e^{-\xi})u_0 = 2(\cosh\xi)u_0$$

(2.3.2)

$$= (2 + h^2\xi^2)u_0 + O(h^4) ,$$

and

$$2u_2 = 2e^{\eta}u_0$$

(2.3.3) $$= (2 + 2h\eta + h^2\eta^2 + \frac{h^3}{3}\eta^3)u_0 + O(h^4)$$

$$= 2u_0 + 2h\gamma(x) - h^2\xi^2u_0 - \frac{h^3}{3}\gamma''(x) + O(h^4) .$$

In (2.3.3), we have used the facts that

$$\eta u_0 = \gamma(x) ,$$

$$\eta^2 u_0 = -\xi^2 u_0 ,$$

and

$$\eta^3 u_0 = -\xi^2 \eta u_0 = -\xi^2 \gamma(x) = \gamma''(x) \ .$$

Adding equations (2.3.2) and (2.3.3), we obtain (2.3.1).

Q.E.D.

Corollary 2.1. *Under the assumptions of Theorem 2.6 with* $\gamma(x) = 0$ *on* L *, an approximation to Laplace's equation at the boundary point 0 (Figure 2.4) is given by*

$$4u_0 = u_1 + 2u_2 + u_3 + O(h^4) \ . \tag{2.3.4}$$

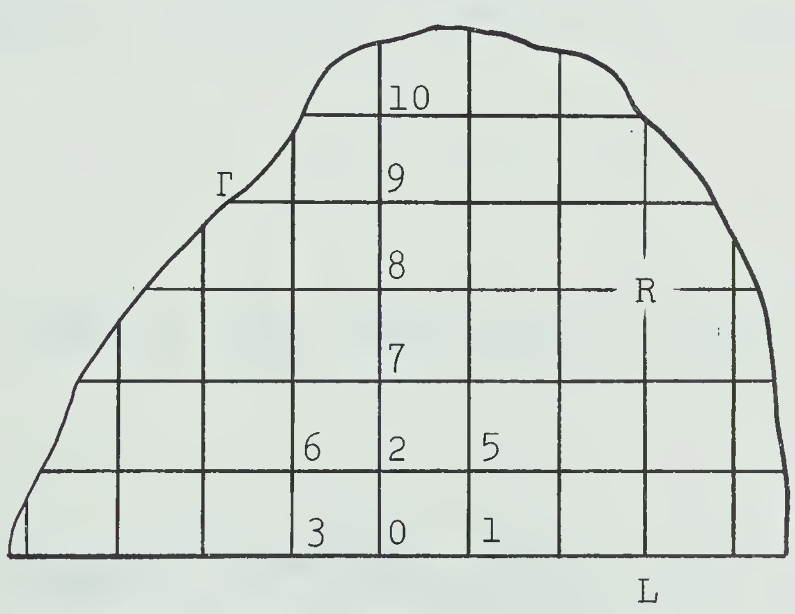

Figure 2.4 Normal-Gradient Specified Along a Straight Line

Formula (2.3.4) is equivalent to a five-point formula (2.2.3) if it is assumed that the solution u is reflected about the line L. For this reason, formula (2.3.4), is sometimes known as the mirror image.

If u is not harmonic on L, then (2.3.1) is not a valid formula. A formula which is free of this restriction is given by

<u>Theorem 2.7</u> *Let $u = u(x,y)$ be harmonic in a region R with boundary Γ. Let the lower part of the boundary Γ be a straight line L parallel to the x-axis. Assume that u has a Taylor's series' expansion about the boundary point 0 (Figure 2.4) with radius of convergence of at least $3h$, and that, $\frac{\partial u}{\partial y} = \gamma(x)$ on L. Then an approximation to Laplace's equation at the point 0 is given by*

$$(2.3.5) \qquad 11u_0 = 18u_2 - 9u_7 + 2u_8 - 6h\gamma(x) + O(h^4) \; .$$

<u>Proof</u>: Expanding u_2, u_4, and u_5 by Taylor's series about the point 0, we obtain

$$
\begin{aligned}
18u_2 - 9u_7 + 2u_8 &= (18e^{\eta} - 9e^{2\eta} + 2e^{3\eta})u_0 \\
&= \{18(1 + h\eta + \frac{h^2}{2}\,\eta^2 + \frac{h^3}{3!}\,\eta^3) \\
&\quad - 9(1 + 2h\eta + \frac{4h^2}{2}\,\eta^2 + \frac{8h^3}{3!}\,\eta^3) \\
&\quad + 2(1 + 3h\eta + \frac{9h^2}{2}\,\eta^3 + \frac{27h^3}{3!}\,\eta^3)\}u_0 \\
&\quad + O(h^4) \\
&= \{11 + 6h\eta\}u_0 + O(h^4) \ .
\end{aligned}
\tag{2.3.6}
$$

Substituting

$$\eta u_0 = \gamma(x)$$

in (2.3.6), we obtain formula (2.3.5).

Q.E.D.

The same methods can be used to derive higher-order approximations to Laplace's equation on boundary points where the boundary is a straight line and the normal-gradient is specified. Approximations of $O(h^6)$ are thus stated without proof.

Theorem 2.8 *Let all the conditions of Theorem 2.6 be satisfied, except now the radius of convergence for the Taylor's series' expansion about the boundary point 0 (Figure 2.4) must be at least* $h\sqrt{2}$. *Then an approximation to Laplace's equation at the point 0 is given by*

$$10u_0 = 2u_1 + 4u_2 + 2u_3 + u_5 + u_6$$

$$- 6h\gamma(x) - \frac{h^5}{30}\gamma^{(4)}(x) \tag{2.3.7}$$

$$+ O(h^6) .$$

Corollary 2.2. *Under the assumptions of Theorem 2.8 with* $\gamma(x) = 0$ *on* L , *the 'mirror-image' approximation to Laplace's equation at the point 0 is given by*

$$10u_0 = 2u_1 + 4u_2 + 2u_3 + u_5 + u_6 + O(h^6) \; . \tag{2.3.8}$$

Theorem 2.9. *Let all the conditions of Theorem 2.7 be satisfied, except now the radius of convergence for the Taylor's series' expansion about the boundary point 0 (Figure 2.4) must be at least* $5h$. *Then an approximation to Laplace's equation at the point 0 is given by*

$$\begin{aligned} 137u_0 &= 300u_2 - 300u_7 + 200u_8 - 75u_9 + 12u_{10} \\ &\quad + 60h\gamma(x) + O(h^6) \; . \end{aligned} \tag{2.3.9}$$

2.3.2 Normal-Gradient Specified Along a Curved Boundary

When the boundary Γ for the Neumann problem is curved, the problem of deriving suitable formulae for points near the boundary is much more formidable. Two main methods used are due to Fox [9] and Viswanathan [22]. These methods introduce truncation errors which are greater than that of the five-point formula; so that, in applying the methods, the interval size should be decreased.

Fox introduces external points like 1 and 4 in Figure 2.5

in the standard five-point formula (2.2.3) relevant to the point 0. The external values are then related, through the normal derivative condition, with internal values, and therefore eliminated from the five-point formula.

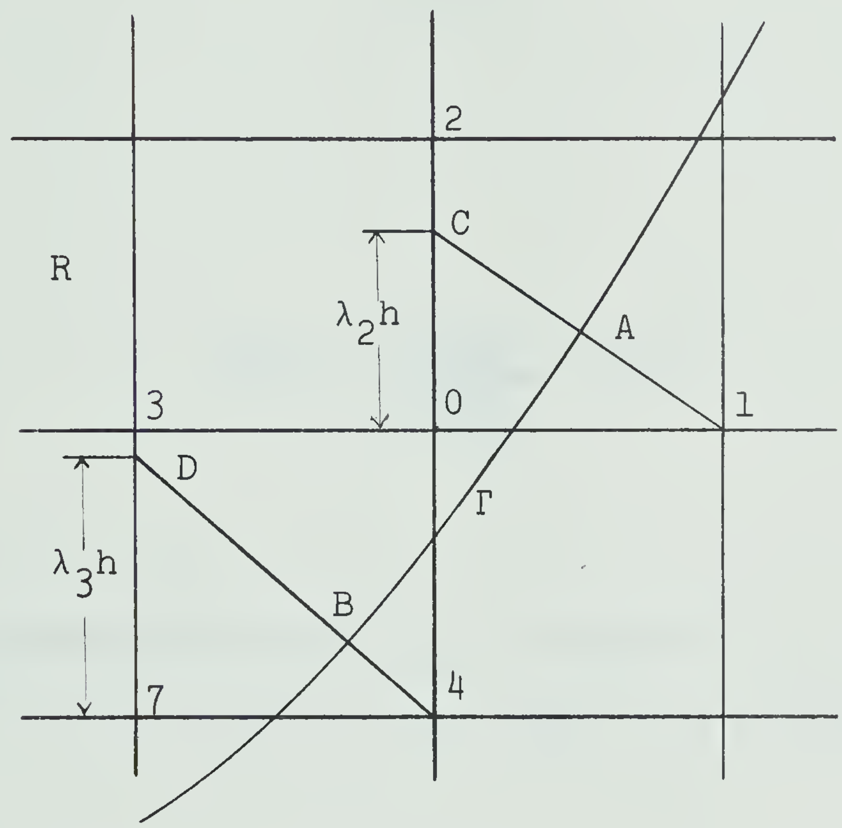

Figure 2.5 Normal-Gradient Specified Along a Curved Boundary

By drawing normals from the point 1 to the boundary Γ at A, and from point 4 to Γ at B, so that they meet internal mesh lines at C and D, respectively, we observe the typical case shown in Figure 2.5. If $\lambda_2 h$

denotes the distance from points 0 to C and $\lambda_3 h$ the distance from points 7 to D, we have that

(2.3.10) $$\gamma(A) = \frac{u_1 - u_C}{h\sqrt{\lambda_2^3 + 1}} + O(h) ,$$

and

(2.3.11) $$\gamma(B) = \frac{u_4 - u_D}{h\sqrt{\lambda_3^2 + 1}} + O(h) .$$

Linear interpolation for u_C in terms of u_2 and u_0, and for u_D in terms of u_3 and u_7, yields

(2.3.12) $$u_C = \lambda_2 u_2 + (1-\lambda_2) u_0 + O(h^2) ,$$

and

(2.3.13) $$u_D = \lambda_3 u_3 + (1-\lambda_3) u_7 + O(h^2) ,$$

respectively. Substituting (2.3.12) and (2.3.13) into equations (2.3.10) and (2.3.11), respectively, we obtain

$$(2.3.14) \qquad u_1 = \lambda_2 u_2 + (1-\lambda_2)u_0 + h\sqrt{\lambda_2^2+1}\ \gamma(A) + O(h^2) \ ,$$

and

$$(2.3.15) \qquad u_4 = \lambda_3 u_3 + (1-\lambda_3)u_7 + h\sqrt{\lambda_3^2+1}\ \gamma(B) + O(h^2) \ ,$$

Thus, the five-point formula (2.2.3) can be implemented, at the point 0, but with an increased truncation error of $O(h^2)$.

The second method due to Viswanathan is more complex but secures greater accuracy. The proofs, likewise, are very complicated so that the formulae are stated without proof.

Assume that Γ has a continuously turning tangent; that is, corners are excluded.

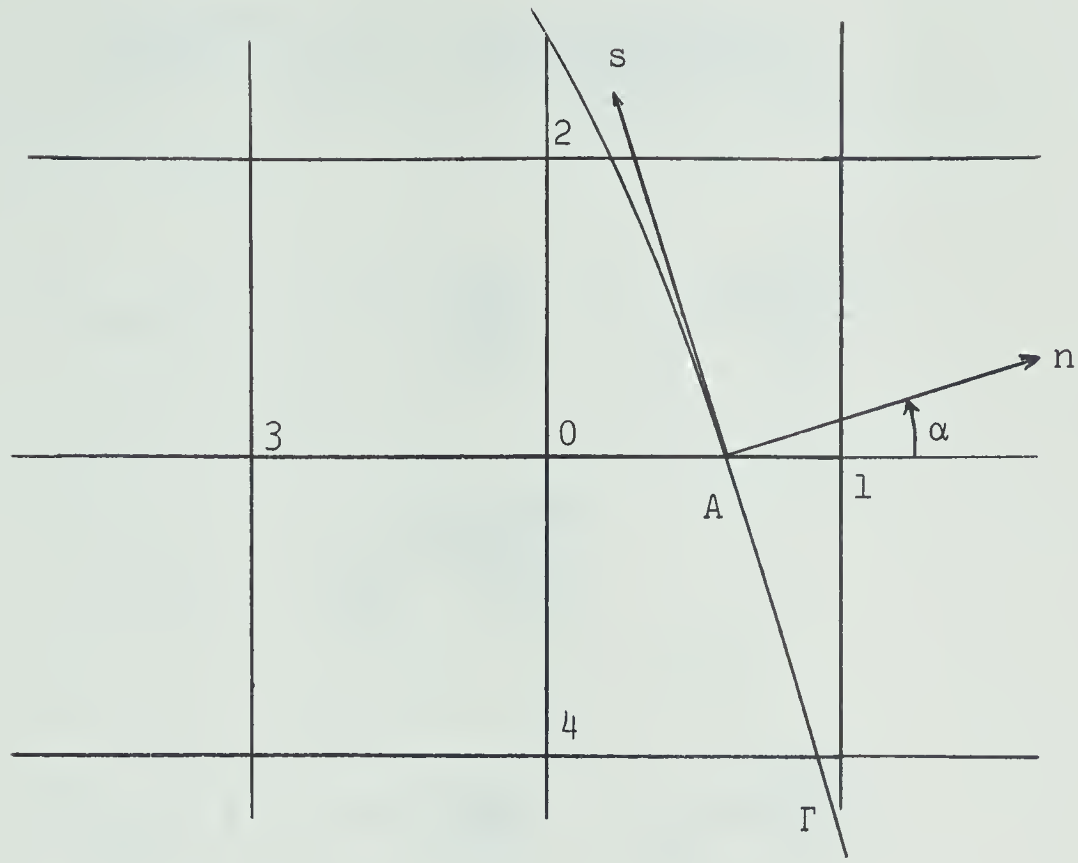

Figure 2.6 Normal-Gradient Specified Along a Curved Boundary

Suppose that, with reference to the five-point formula at the point 0 (Figure 2.6), a single point 1 is external to the region R . If λh denotes the distance from the point 0 to the point A , and if the normal to Γ at the point A makes and angle α with the mesh line, then

$$- 2h\sqrt{1+m^2}\ \gamma(A) + 2\lambda h^2 m \frac{1+m^2}{1-m^2} \left(\frac{\partial\gamma}{\partial s}\right)_A$$

$$= u_2\left(1 - m + 2\lambda \frac{1+m^2}{1-m^2} + \lambda \frac{h}{\rho_A} \frac{m\sqrt{1+m^2}}{1-m}\right)$$

$$+ 2u_3\left(1 + \lambda \frac{h}{\rho_A} \frac{m^2\sqrt{1+m^2}}{1-m^2}\right)$$

(2.3.16)

$$+ u_4\left(1 + m + 2\lambda \frac{1+m^2}{1-m^2} - \lambda \frac{h}{\rho_A} \frac{m\sqrt{1+m^2}}{1-m}\right)$$

$$- 4u_0\left(1 + \lambda \frac{1+m^2}{1-m^2} + \lambda \frac{h}{\rho_A} \frac{m^2\sqrt{1+m^2}}{1-m^2}\right)$$

$$+ O(h^3) ,$$

where

$$m = \tan \alpha ,$$

and ρ_A denotes the radius of curvature of Γ at the point A .

If two points, 1 and 2 as in Figure 2.7, with reference to the five-point formula at the point 0 are external to the region R, then

$$- \sqrt{2}\, h \sqrt{1+m^2}\, \gamma(A) + \lambda h^2 m \frac{1+m^2}{1-m^2} \left(\frac{\partial \gamma}{\partial s}\right)_A$$

$$= u_3\left(-m + \lambda \frac{1+m^2}{1-m^2} + \frac{\lambda h}{\sqrt{2}\rho_A} \frac{m\sqrt{1+m^2}}{1-m^2}\right)$$

$$+ u_7\left(1 - \lambda \frac{1+m^2}{1-m^2} + \frac{\lambda h}{\sqrt{2}\rho_A} \frac{m^2\sqrt{1+m^2}}{1-m^2}\right)$$

(2.3.17)

$$+ u_4\left(m + \lambda \frac{1+m^2}{1-m^2} - \frac{\lambda h}{\sqrt{2}\rho_A} \frac{m\sqrt{1+m^2}}{1-m^2}\right)$$

$$- u_0\left(1 + \lambda \frac{1+m^2}{1-m^2} + \frac{\lambda h}{\sqrt{2}\rho_A} \frac{m^2\sqrt{1+m^2}}{1-m^2}\right)$$

$$+ O(h^3) .$$

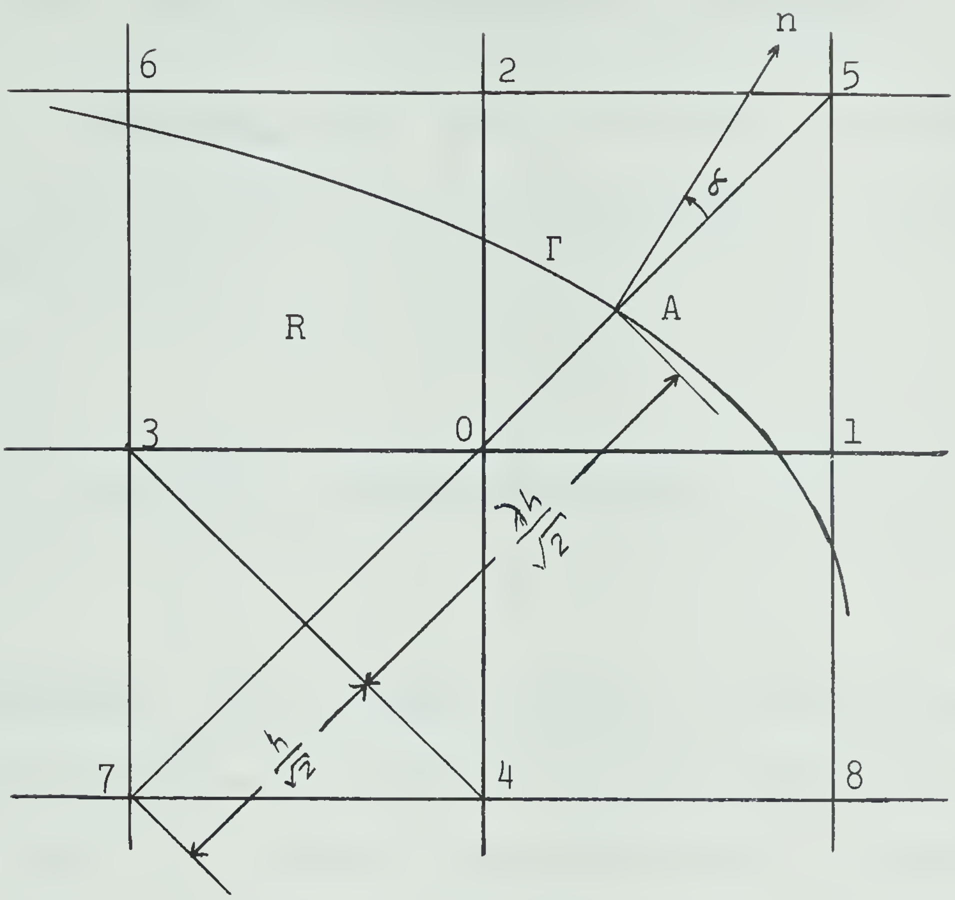

Figure 2.7 Normal-Gradient Specified Along a Curved Boundary

2.4 The Third Boundary-Value Problem

For the third boundary-value problem, standard formulae, such as the five-point or the nine-point formulae, can be used to approximate Laplace's equation for points not near the boundary. For points near the boundary, as is the case for the Neumann problem, special formulae must be developed.

2.4.1 The Third Problem on a Rectangle

Theorem 2.10. *Let* $u = u(x,y)$ *be harmonic in a rectangular region* R *with boundary* Γ *, along which equation (1.1.5),*

$$\alpha(P)u(P) + \beta(P)u_n(P) = \alpha(P) \;, \quad P \text{ on } \Gamma \;,$$

is satisfied. If u *has a Taylor's series' expansion about a point 0 (Figure 2.8) near* Γ *with radius of convergence of at least* $2h$ *, then an approximation to Laplace's equation at the point 0 is given by*

$$(12\alpha_1 - 13\beta_1)u_0 = 3\gamma_1 + (3\alpha_1 - \frac{11}{2}\beta_1)(u_2 + u_4)$$

(2.4.1)

$$+ (3\alpha_1 - \beta_1)u_3 - \beta_1 u_5 + O(h^4) \;.$$

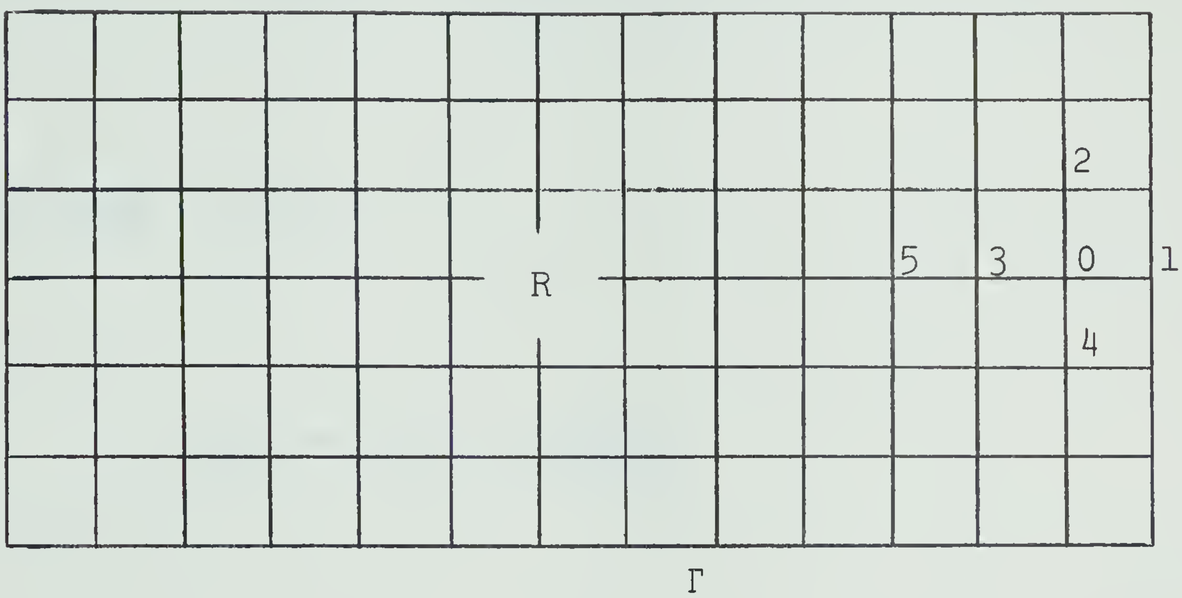

Figure 2.8 The Third Problem on a Rectangle

Proof: An approximation to $\frac{\partial u}{\partial n}$ at the point 1 in terms of u_0, u_1, u_3 and u_5 is obtained from

$$18u_0 - 9u_3 + 2u_5 = (18e^{-h\xi} - 9e^{-2h\xi} + 2e^{-3h\xi})u_1$$

$$= 11u_1 - 6\left(\frac{\partial u}{\partial n}\right)_1 + O(h^4) \; ;$$

that is,

$$(2.4.2) \qquad 6\left(\frac{\partial u}{\partial n}\right)_1 = 18u_0 - 11u_1 - 9u_3 + 2u_5 + O(h^4) .$$

But from (1.1.5),

$$6\alpha_1 u_1 = 6\gamma_1 - 6\beta_1\left(\frac{\partial u}{\partial n}\right)_1$$

$$= 6\alpha_1 - \beta_1(18u_0 - 11u_1 - 9u_3 + 2u_5) + O(h^4) ,$$

so that

$$(2.4.3) \qquad (6\alpha_1 - 11\beta_1)u_1 = 6\gamma_1 - \beta_1(18u_0 - 9u_3 + 2u_5) + O(h^4) .$$

Substitution of (2.4.3) into the five-point formula (2.2.3) gives

$$4(6\alpha_1-11\beta_1) = (6\alpha_1-11\beta_1)(u_1+u_2+u_3+u_4) + O(h^4)$$

$$(2.4.4) \qquad = (6\alpha_1-11\beta_1)(18u_0-9u_3+2u_5)$$

$$+ (6\alpha_1-11\beta_1)(u_2+u_3+u_4) + O(h^4) .$$

Simplification of (2.4.4) yields formula (2.4.1).

Q.E.D.

2.4.2 <u>The Third Problem with a Curved Boundary</u>

Finding formulae for the third boundary-value problem for points near a curved boundary is much more complicated. A formula with truncation error of $O(h)$, similar to Fox's formula for the Neumann problem, is presented.

Suppose that, relative to the five-point formula at the point 0 (Figure 2.9), the point 1 is the only external point. u_1 is then related to internal values by the use of the third boundary condition (1.1.5) and thus eliminated from the five-point formula.

Let δ denote the angle that the normal from the point 1 to the boundary Γ at A makes with the mesh line. Then if λh , $0 \le \lambda < 1$, is the distance from the

point 1 to the point A along the mesh line (see Figure 2.9), we have, on expansion by Taylor's series, that

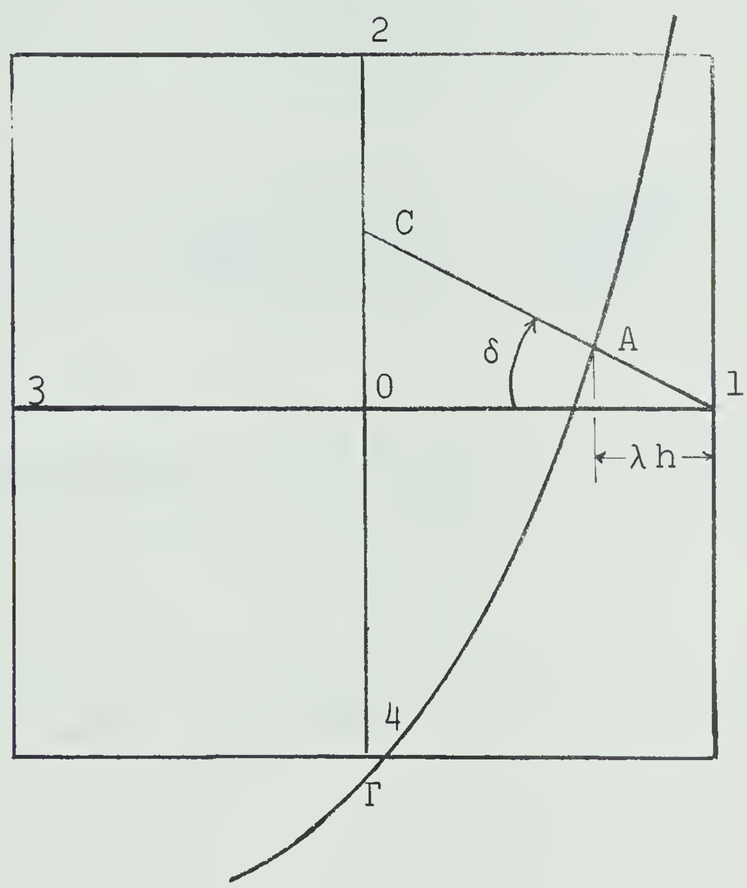

Figure 2.9 The Third Problem with a Curved Boundary

$$u_1 = e^{\frac{\lambda h}{\cos\delta}\frac{\partial}{\partial n}} u_A$$

(2.4.5)

$$= u_A + \frac{\lambda h}{\cos\delta}\left(\frac{\partial u}{\partial n}\right)_A + \frac{\lambda^2 h^2}{2\cos^2\delta}\left(\frac{\partial^2 u}{\partial^2 n}\right)_A + O(h^3) ,$$

and

$$u_C = e^{-\frac{(1-\lambda)h}{\cos\,\delta}\frac{\partial}{\partial n}} u_A$$

(2.4.6)

$$= u_A - \frac{(1-\lambda)h}{\cos\,\delta} \left(\frac{\partial u}{\partial n}\right)_A + \frac{(1-\lambda)^2 h^2}{\cos^2 \delta} \left(\frac{\partial^2 u}{\partial n^2}\right)_A + O(h^3) \; .$$

Multiplying equations (2.4.5) and (2.4.6) by

(2.4.7) $$k_1 = \frac{1}{h} \left(\alpha_A h(1-\lambda) + \beta_A \cos\,\delta\right) \, ,$$

and

(2.4.8) $$k_2 = \frac{1}{h} \left(\alpha_A \lambda h - \beta_A \cos\,\delta\right) \, ,$$

respectively, and adding, we obtain

$$k_1u_1 + k_2u_C = \alpha_A u_A + \beta_A \left(\frac{\partial u}{\partial n}\right)_A$$

$$(2.4.9) \qquad + \frac{h}{2\cos\delta}\left[\frac{\lambda(1-\lambda)h\alpha_A}{\cos\delta} + (2\lambda-1)\beta_A\right]\left(\frac{\partial^2 u}{\partial n^2}\right)_A$$

$$+ O(h^3) .$$

Since at the point A, (1.1.5) is true, equation (2.4.9) yields the following approximation to u_1 in terms of interior points:

$$(2.4.10) \qquad k_1u_1 = \gamma_A - k_2u_C + O(h) .$$

Linear interpolation for u_C in terms of u_0 and u_2 yields

$$(2.4.11) \qquad u_C = \tan\delta\, u_2 + (1-\tan\delta)u_0 + O(h^2) ,$$

which, when substituted into (2.4.10), gives

(2.4.12) $$k_1 u_1 = \gamma_A - k_2[\tan\delta\ u_2 + (1-\tan\delta)u_0] + O(h) .$$

Equation (2.4.12) is now substituted in the five-point formula to produce the final form

(2.4.13)
$$\begin{aligned}[4k_1 - (1-\tan\delta)k_2]u_0 &= \gamma_A + k_1(u_3+u_4) \\ &\quad + (k_1-\tan\delta\ k_2)u_2 + O(h) .\end{aligned}$$

Since the truncation error is so large in formula (2.4.5), its application would require the use of a smaller interval size at the boundary. A more general formula, for which two points relevant to the five-point formula are exterior to the region under consideration, can similarly be derived.

CHAPTER III

TORSION OF A CROSS-SECTIONED GIRDER

3.1 Statement of the Problem

The torsion problem for a beam of square cross-section (with sides of length 2) leads to the following Dirichlet problem (see Collatz [4]):

Problem 3.1 *Determine* $u = u(x,y)$ *in the region* R *with boundary* Γ *(Figure 3.1) so that*

$$\nabla^2 u = 0 \quad \text{in} \quad R , \tag{3.1.1}$$

and

$$u = x^2 + y^2 \quad \text{on} \quad \Gamma . \tag{3.1.2}$$

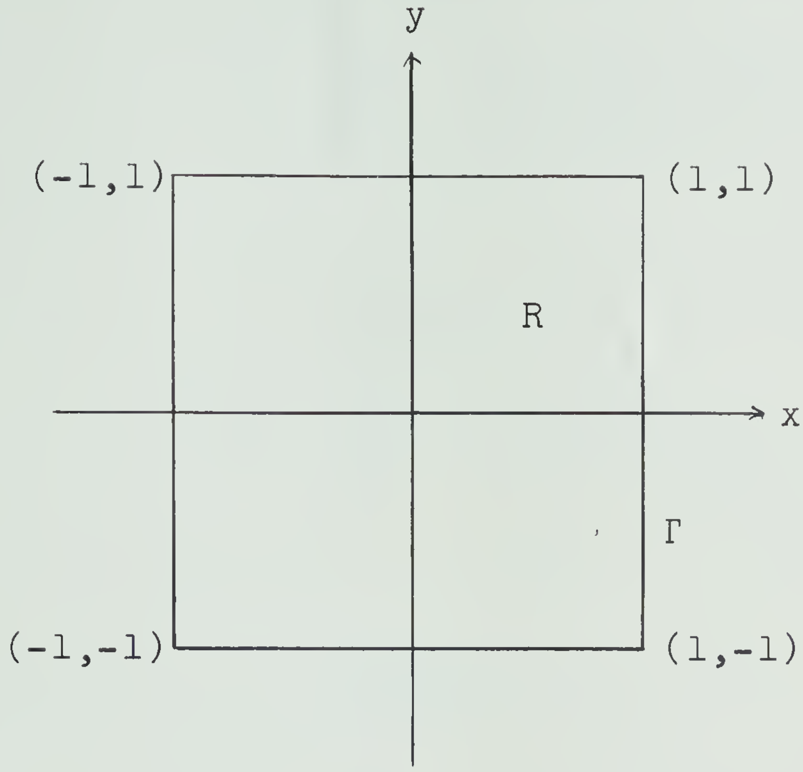

Figure 3.1 The Torsion Problem

An analytical solution to Problem 3.1 can be obtained by considering four sets of boundary conditions, which may be thought of as representing four cases of the steady-state temperature distribution in a square plate (see Hildebrand [12]), where the region R now represents a square plate. If we solve for u_i $(i=1,2,3,4)$, u_i harmonic in R and with boundary conditions:

(3.1.3)
$$u_1 = \begin{cases} 1+x^2 & \text{on } \Gamma_1 , \\ 0 & \text{on } \Gamma_2,\Gamma_3,\Gamma_4 ; \end{cases}$$

(3.1.4)
$$u_2 = \begin{cases} 1+x^2 & \text{on } \Gamma_2 , \\ 0 & \text{on } \Gamma_1,\Gamma_3,\Gamma_4 ; \end{cases}$$

(3.1.5)
$$u_3 = \begin{cases} 1+y^2 & \text{on } \Gamma_3 , \\ 0 & \text{on } \Gamma_1,\Gamma_2,\Gamma_4 ; \end{cases}$$

and

(3.1.6)
$$u_4 = \begin{cases} 1+y^2 & \text{on } \Gamma_4 , \\ 0 & \text{on } \Gamma_1,\Gamma_2,\Gamma_3 ; \end{cases}$$

where Γ_i are the boundary segments defined by:

$$\Gamma_1: \quad y=1 \;, \quad -1 \le x \le 1 \;;$$

$$\Gamma_2: \quad y=-1, \quad -1 \le x \le 1 \;;$$

$$\Gamma_3: \quad x=1 \;, \quad -1 \le y \le 1 \;;$$

$$\Gamma_4: \quad x=-1, \quad -1 \le y \le 1 \;;$$

then the sum of the solutions,

$$u = \sum_{i=1}^{4} u_i \;, \tag{3.1.7}$$

is the solution to Problem 3.1. This follows from the fact that the sum of two harmonic functions is harmonic, and that the sum of the boundary conditions (3.1.3) - (3.1.6) yields the boundary condition (3.1.2) on $\Gamma = \Gamma_1 \cup \Gamma_2 \cup \Gamma_3 \cup \Gamma_4$.

The solutions of the four cases (3.1.3) - (3.1.6) can be determined by the method of separation of variables [12], which, on addition, yield the infinite series solution,

$$u(x,y) = \sum_{k=1}^{\infty} c_k \{\sin d_k(x+1) \cdot \cosh d_k y$$

(3.1.8)

$$+ \sin d_k(y+1) \cdot \cosh d_k x\} ,$$

where

(3.1.9) $$c_k = \frac{8}{\pi^3(2k-1)} (\pi + \frac{2}{2k-1})(\pi - \frac{2}{2k-1}) \frac{1}{\cosh d_k}$$

and

(3.1.10) $$d_k = \frac{(2k-1)}{2} \pi .$$

The series (3.1.8) may or may not be convergent in all parts of the region of interst; and, even if it is convergent, evaluation of the series by summing a finite number of terms may require more calculations than required when solving the problem numerically.

In solving Problem 3.1 numerically, we observe that the problem is symmetric about the origin so that only the region in the first quadrant need be considered. Suppose that n mesh lines in both the x and y directions

are necessary to cover the simplified region R^* (Figure 3.2) by a square grid-system of mesh length h. Approximation of Laplace's equation by finite-difference formulae at the interior gridpoints

$$(i,j)\ ,\quad \begin{aligned} j&=1,2,\ldots,n\ , \\ i&=1,2,\ldots,n\ , \end{aligned} \tag{3.1.11}$$

then reduces the problem to one of solving a system of linear algebraic equations.

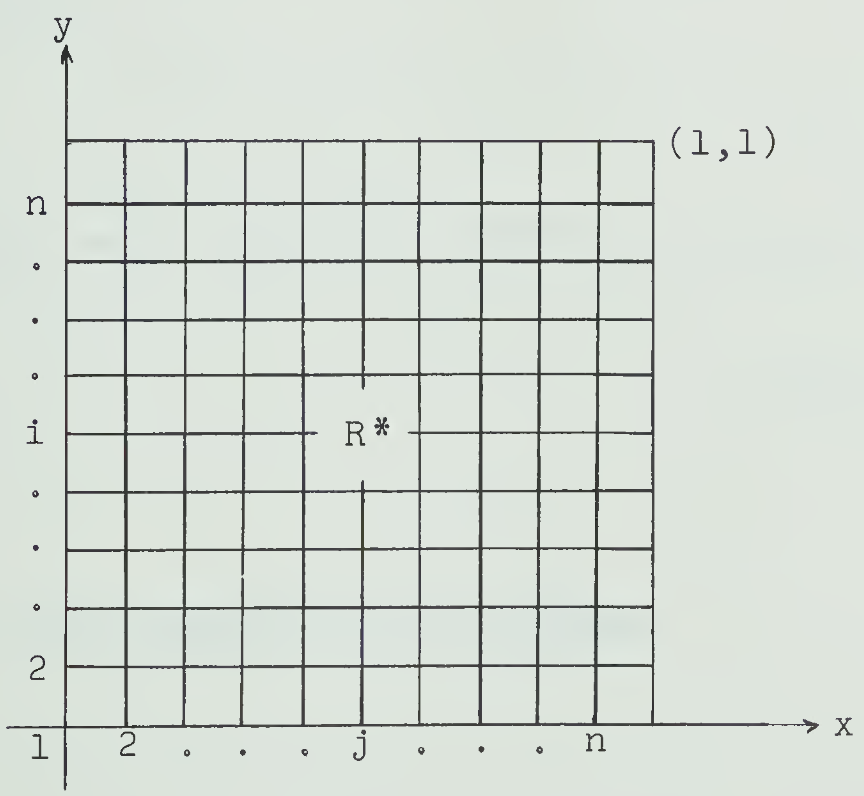

Figure 3.2 Grid-System for the Torsion Problem

3.2 Solution by Use of the Five-Point Formula

We assume that $nh = 1$ in Figure 3.2, so that the boundary condition (3.1.2) yields

$$u(n+1,i) = u(i,n+1) = 1+(i-1)^2h^2 , \qquad i=1,2,\ldots,n+1 , \tag{3.2.1}$$

for gridpoints on the boundary Γ. For the interior gridpoints (3.1.11), the five-point formula (2.2.3) can be written as

$$u(i,j) = \frac{1}{4} [u(i,j+1) + u(i+1,j) + u(i,j-1) + u(i-1),j)] , \qquad i,j=2,3,\ldots,n ; \tag{3.2.2}$$

and from the symmetry of the problem

$$u(1,1) = \frac{1}{2}\,[u(1,2) + u(2,1)] \;;$$

$$u(1,j) = \frac{1}{4}\,[u(1,j-1) + 2u(2,j) + u(1,j+1)] \;,$$

$$(3.2.3) \qquad\qquad j=2,3,\ldots,n \;;$$

$$u(i,1) = \frac{1}{4}\,[u(i-1,1) + 2u(i,2) + u(i+1,1)] \;,$$

$$i=2,3,\ldots,n \;.$$

Let I, J and D_k be matrices of order n, where I is the identity matrix,

$$(3.2.4) \qquad\qquad J = 2I \;,$$

and

$$(3.2.5) \qquad D_k = \begin{bmatrix} -4 & 2 & & & & & \\ 1 & -4 & 1 & & & & \\ & 1 & -4 & 1 & & & \\ & & & \cdot & & & \\ & & & & \cdot & & \\ & & & & & \cdot & \\ & & & & 1 & -4 & 1 \\ & & & & & 1 & -4 \end{bmatrix}, \quad k=1,2,\ldots,n .$$

In (3.2.5), the elements with unassigned values are zero. If B_k and U_k are vectors of dimension n ,

$$B_k = [0,\ 0,\ \ldots,\ 0,\ 1+(k-1)^2h^2]^T , \quad k=1,2,\ldots,n-1 ,$$

(3.2.6)

$$B_n = [1,\ 1+h^2,\ \ldots,\ 1+(n-2)^2h^2,\ 2+2(n-1)^2h^2]^T ;$$

and

$$(3.2.7) \quad U_k = [u(k,1),u(k,2),\ldots,u(k,n)]^T , \quad k=1,2,\ldots,n ;$$

then the system (3.2.2) and (3.2.3) can be written in matrix notation as

$$(3.2.8) \qquad AU = B \ ,$$

where A is the block, tridiagonal matrix,

$$(3.2.9) \qquad A = \begin{bmatrix} D_1 & J & & & & & \\ I & D_2 & I & & & & \\ & I & D_3 & I & & & \\ & & & \cdot & & & \\ & & & & \cdot & & \\ & & & & & \cdot & & \\ & & & & I & D_{n-1} & I \\ & & & & & I & D_n \end{bmatrix} ;$$

and U and B are the block vectors defined by

$$(3.2.10) \qquad U = [U_1^T, U_2^T, \ldots, U_n^T]^T \ ,$$

and

$$(3.2.11) \qquad B = -[B_1^T, B_2^T, \ldots, B_n^T]^T \ .$$

The system (3.2.8) can be solved by the point-Gauss-Seidel method which consists of recursively applying equations (3.2.2) and (3.2.3) to the interior gridpoints of the simplified region R^* . A consistent ordering of gridpoints (one such ordering is given by (3.1.11)) must first, of course, be chosen. The point-Gauss-Seidel method requires approximately $(n+1)^2$ multiplications and $3(n+1)^2$ additions for each iteration.

When solving the system (3.2.8) by the Gaussian elimination method, we first decompose A into a lower-triangular matrix L and an upper-triangular matrix P , so that

$$A = LP \ . \tag{3.2.12}$$

Forward-elimination reduces (3.2.8) to

$$PU = L^{-1}B \tag{3.2.13}$$

from which U can be computed by back-substitution.

It can be seen from (3.2.9) that decomposition of A can be performed by blocks; that is, elimination in the k'th block effects the elements in blocks k and $k+1$

only. For example, when decomposing the k'th block $(1<k<6)$ for $n=6$, elimination is performed on

$$
(3.2.14) \qquad \left[\begin{array}{c|c} D_k^* & I \\ \hline I & D_{k+1} \end{array}\right] = \left[\begin{array}{cccccc|cccccc}
x & x & x & x & x & x & 1 & & & & & \\
x & x & x & x & x & x & & 1 & & & & \\
x & x & x & x & x & x & & & 1 & & & \\
x & x & x & x & x & x & & & & 1 & & \\
x & x & x & x & x & x & & & & & 1 & \\
x & x & x & x & x & x & & & & & & 1 \\
\hline
1 & & & & & & -4 & 2 & & & & \\
 & 1 & & & & & 1 & -4 & 1 & & & \\
 & & 1 & & & & & 1 & -4 & 1 & & \\
 & & & 1 & & & & & 1 & -4 & 1 & \\
 & & & & 1 & & & & & 1 & -4 & 1 \\
 & & & & & 1 & & & & & 1 & -4
\end{array}\right]
$$

where D_k^*, with elements x of arbitrary value, denotes the new D_k resulting from the elimination of block $k-1$. At the third stage of the elimination of (3.2.14), we have

$$
\left[\begin{array}{cccccc|cccccc}
x & x & x & x & x & x & 1 & & & & & \\
 & x & x & x & x & x & x & 1 & & & & \\
 & & x & x & x & x & x & x & & & & \\
 & & x & x & x & x & x & x & & & & \\
 & & x & x & x & x & x & x & & & & \\
 & & x & x & x & x & x & x & & & & \\
\hline
 & & x & x & x & x & x & x & & & & \\
 & & x & x & x & x & x & x & 1 & & & \\
 & & 1 & & & & & 1 & -4 & 1 & & \\
 & & & 1 & & & & & 1 & -4 & 1 & \\
 & & & & 1 & & & & & 1 & -4 & 1 \\
 & & & & & 1 & & & & & 1 & -4
\end{array}\right];
$$

Figure 3.3

and after the final elimination, we have

$$
\left[\begin{array}{cccccc|cccccc}
x & x & x & x & x & x & 1 & & & & & \\
 & x & x & x & x & x & x & 1 & & & & \\
 & & x & x & x & x & x & x & 1 & & & \\
 & & & x & x & x & x & x & x & 1 & & \\
 & & & & x & x & x & x & x & x & 1 & \\
 & & & & & x & x & x & x & x & x & 1 \\
\hline
 & & & & & & x & x & x & x & x & x \\
 & & & & & & x & x & x & x & x & x \\
 & & & & & & x & x & x & x & x & x \\
 & & & & & & x & x & x & x & x & x \\
 & & & & & & x & x & x & x & x & x \\
 & & & & & & x & x & x & x & x & x
\end{array}\right],
$$

Figure 3.4

where the lower right-hand block in Figure 3.4 is now D^*_{k+1} .

We have assumed, in Figures 3.3 and 3.4, that no pivoting was performed during the elimination process. This, however, is a good assumption since A is diagonally dominant and pivoting is unnecessary unless n is very large; and even then, pivoting is unnecessary until the final stages of the elimination.

Since the decomposed matrix A (the upper-triangular matrix P in (3.2.13)) is of band form with a band width of n (assuming that pivoting is not used and the 1's on top of the band (see Figure 3.4) are ignored), a matrix of dimension $n^2 \times n$ is large enough to store all the results of the forward-elimination of A . Approximately n^4 multiplications and the same number of additions are required for the forward-elimination and back-substitution processes.

3.3 Solution by Use of the Nine-Point Formula

When the nine-point formula (2.2.16) is used to approximate Laplace's equation at the interior gridpoints (3.1.11), the problem is reduced to solving the following system of linear algebraic equations:

[illegible]

[illegible]

$$u(i,j) = \frac{1}{20} \{4[u(i,j+1) + u(i-1,j) + u(i,j-1) + u(i+1,j)]$$

$$(3.3.1) \qquad + u(i-1,j+1) + u(i-1,j-1) + u(i+1,j-1) + u(i+1,j+1)\},$$

$$i,j=2,3,\ldots,n \;;$$

and from the symmetry of the problem,

$$u(1,1) = \frac{1}{5} \{2[u(1,2) + u(2,1)] + u(2,2)\} \;;$$

$$u(1,j) = \frac{1}{10} \{2u(1,j+1) + u(2,j+1) + 4u(2,j)$$

$$+ u(2,j-1) + 2u(1,j-1)\} \;,$$

$$(3.3.2) \qquad j=2,3,\ldots,n \;;$$

$$u(i,1) = \frac{1}{10} \{2u(i-1,1) + u(i-1,2) + 4u(i,2)$$

$$+ u(i+1,2) + 2u(i+1,1)\} \;,$$

$$i=2,3,\ldots,n \;;$$

given that, the boundary condition (3.2.1) holds true along the boundary Γ .

Let L and D_k be matrices of order n defined by

$$(3.3.3) \qquad L = \begin{bmatrix} 4 & 2 & & & & & \\ 1 & 4 & 1 & & & & \\ & 1 & 4 & 1 & & & \\ & & & \cdot & & & \\ & & & & \cdot & & \\ & & & & & \cdot & \\ & & & & 1 & 4 & 1 \\ & & & & & 1 & 4 \end{bmatrix},$$

$$(3.3.4) \qquad D_1 = \begin{bmatrix} -10 & 4 & & & & & \\ 2 & -10 & 2 & & & & \\ & 2 & -10 & 2 & & & \\ & & & \cdot & & & \\ & & & & \cdot & & \\ & & & & & \cdot & \\ & & & & 2 & -10 & 2 \\ & & & & & 2 & -10 \end{bmatrix},$$

and

$$(3.3.5) \qquad D_k = 2D_1 \ , \quad k=2,3,\ldots,n \ .$$

If B_k is a vector of dimension n defined by

$$B_1 = [0,0,\ldots,0,\tfrac{1}{2}\, b_1]^T ,$$

(3.3.6) $$B_k = [0,0,\ldots,0,b_k]^T, \quad k=2,3,\ldots,n-1 ,$$

$$B_n = [b_1,b_2,\ldots,b_{n-1},2b_n]^T ,$$

and

(3.3.7) $$b_k = 6 + (6k^2 - 12k + 8)h^2 , \quad k=1,2,\ldots,n ;$$

then the system (3.3.1) and (3.3.2) can be written in matrix form as in (3.2.8), with

$$(3.3.8) \qquad A = \begin{bmatrix} D_1 & L & & & & & \\ L & D_2 & & & & & \\ & L & D_3 & L & & & \\ & & & \cdot & & & \\ & & & & \cdot & & \\ & & & & & \cdot & \\ & & & & L & D_{n-1} & L \\ & & & & & L & D_n \end{bmatrix},$$

and U and B defined as in (3.2.10) and (3.2.11), respectively.

As for the five-point formula, decomposition of A by the Gaussian elimination method can be performed in blocks, so that for the k'th block (with $n=6$), elimination is performed on

(3.3.9)

$$
\left[\begin{array}{c|c} D_k^* & L \\ \hline L & D_{k+1} \end{array}\right] =
\left[\begin{array}{cccccc|cccccc}
x & x & x & x & x & x & 4 & 2 & & & & \\
x & x & x & x & x & x & 1 & 4 & 1 & & & \\
x & x & x & x & x & x & & 1 & 4 & 1 & & \\
x & x & x & x & x & x & & & 1 & 4 & 1 & \\
x & x & x & x & x & x & & & & 1 & 4 & 1 \\
x & x & x & x & x & x & & & & & 1 & 4 \\
\hline
4 & 2 & & & & & -20 & 8 & & & & \\
1 & 4 & 1 & & & & 4 & -20 & 4 & & & \\
 & 1 & 4 & 1 & & & & 4 & -20 & 4 & & \\
 & & 1 & 4 & 1 & & & & 4 & -20 & 4 & \\
 & & & 1 & 4 & 1 & & & & 4 & -20 & 4 \\
 & & & & 1 & 4 & & & & & 4 & -20
\end{array}\right]
$$

At the third stage of the elimination of (3.3.9), we have, on assuming that no pivoting is required,

$$
\left[\begin{array}{cccccc|cccccc}
x & x & x & x & x & x & 4 & 2 & & & & \\
 & x & x & x & x & x & x & x & 1 & & & \\
 & & x & x & x & x & x & x & x & 1 & & \\
 & & x & x & x & x & x & x & x & 4 & 1 & \\
 & & x & x & x & x & x & x & x & 1 & 4 & 1 \\
 & & x & x & x & x & x & x & x & & 1 & 4 \\
\hline
 & & x & x & x & x & x & x & x & & & \\
 & & x & x & x & x & x & x & x & & & \\
 & & x & x & x & x & x & x & x & 4 & & \\
 & & 1 & 4 & 1 & & & & 4 & -20 & 4 & \\
 & & & 1 & 4 & 1 & & & & 4 & -20 & 4 \\
 & & & & 1 & 4 & & & & & 4 & -20
\end{array}\right] ;
$$

Figure 3.5

and after the final stage of the elimination, (3.3.9) becomes

$$
\left[\begin{array}{cccccc|cccccc}
x & x & x & x & x & x & 4 & 2 & & & & \\
 & x & x & x & x & x & x & x & 1 & & & \\
 & & x & x & x & x & x & x & x & 1 & & \\
 & & & x & x & x & x & x & x & x & 1 & \\
 & & & & x & x & x & x & x & x & x & 1 \\
 & & & & & x & x & x & x & x & x & x \\
\hline
 & & & & & & x & x & x & x & x & x \\
 & & & & & & x & x & x & x & x & x \\
 & & & & & & x & x & x & x & x & x \\
 & & & & & & x & x & x & x & x & x \\
 & & & & & & x & x & x & x & x & x \\
 & & & & & & x & x & x & x & x & x
\end{array}\right]
$$

Figure 3.6

The decomposed matrix A , as in the case of the five-point formula, is also of band form (see Figure 3.6), but its width is now $n+2$, so that a matrix of dimension $n^2 \times (n+1)$ is required to store the results of the forward-elimination. Furthermore, the number of multiplications, as well as additions, required to obtain the solution of the system resulting from the nine-point formula is increased to approximately $n^2(n+1)^2$.

Solution of the system by the point-Gauss-Seidel method, which consists of recursively applying formulae

(3.3.1) and (3.3.2) to the gridpoints (3.1.11) requires approximately $2n^2$ multiplications and $7n^2$ additions for each iteration.

Comparative computer times required to obtain solutions to the problem by the point-Gauss-Seidel method and the Gaussian elimination method are given in the next section.

3.4 Numerical Results

3.4.1 Presentation of Results Computer programs written for the Fortran IV compiler for the IBM 360/67 computing system at the University of Alberta, Edmonton, Alberta to solve Problem 3.1 by the methods discussed in Sections 3.2 and 3.3 can be found in Appendix II.

A summary of the numerical results for mesh sizes of $\frac{1}{4}$, $\frac{1}{5}$, $\frac{1}{10}$, and $\frac{1}{20}$ is presented in Tables 3.1 and 3.2. A smaller mesh size is not used because of the limited storage capacity of the computer. For example, if the mesh size h is reduced to $\frac{1}{40}$, then a matrix of dimension 1600×41 (65,600 words of storage) is necessary to store the results of the forward-elimination of the matrix A in (3.3.8).

Two types of errors are inherent in solving partial differential equations by finite-difference methods. The first error is due to the truncation error in approximating

the partial differential equation by a finite-difference formula, while the second is the error incurred in solving the resulting system of equations.

In order to determine the truncation error of a finite-difference formula using a particular mesh size h , the system resulting from this approximation must first be solved to a degree of accuracy which is better than the accuracy of the finite-difference formula. This insures that the error of the finite-difference formula is the dominating one. In the ideal situation, the solution at the gridpoints thus calculated is then compared with the exact solution of the problem. When the exact solution is not available, however, an approximation of the truncation error of the finite-difference formula, for that particular mesh size, can be found by comparing the solution obtained with a solution which is significantly more accurate. A more accurate solution can be obtained by using a finite-difference formula with a truncation error of higher order, or by decreasing the interval size. The grid-system must, of course, be chosen so that comparisons can be made at all the gridpoints of the lower-order approximation.

In Tables 3.1 and 3.2, the maximum truncation error in approximating Laplace's equation by the five-point and the nine-point formulae for any particular mesh size is

obtained by first solving the resulting system of equations by the point-Gauss-Seidel method to an accuracy of 5×10^{-6}. The solutions at the grid-systems (corresponding to different mesh sizes) thus obtained are then compared with the solution secured when approximating Laplace's equation by the nine-point formula with a mesh size of $\frac{1}{40}$, and solving the resulting system of equations by the point-Gauss-Seidel method to an accuracy of 5×10^{-6}.

That a solution to the system (3.2.8) obtained by the point-Gauss-Seidel method has an accuracy of e ($e = 5 \times 10^{-6}$ in the above case), means that the maximum deviation of u from the previous iteration is less than or equal to e; or,

(3.4.1) $$\max_{i,j} |u^{k+1}(i,j) - u^{k}(i,j)| < e \; ,$$

where k represents the number of times that the point-Gauss-Seidel iteration has been applied. To begin the iteration, an inital guess $u^0(i,j)$, which for this investigation is always zero, to the solution u must first, of course, be made.

The maximum error in the solution of the system in each case is obtained by first solving the system by the point-Gauss-Seidel method to an accuracy of 5×10^{-6} and

Mesh Size $h=\frac{1}{n}$	Number of Interior Gridpoints	Max. Trun. Error in Five-Point Formula	Max. Error in Solution of System	Time in Secs by Gauss Elim. t_{G5}	Solution by Point-Gauss-Seidel		$\frac{t_{S5}}{t_{G5}}$
					Time in Secs t_{S5}	Number of Iterations	
$\frac{1}{4}$	16	1.7×10^{-2}	1.0×10^{-6}	0.007	0.038	81	5.43
$\frac{1}{5}$	25	8.9×10^{-3}	5.7×10^{-6}	0.015	0.076	108	5.07
$\frac{1}{10}$	100	5.4×10^{-3}	1.1×10^{-4}	0.168	0.689	253	4.10
$\frac{1}{20}$	400	3.7×10^{-3}	3.2×10^{-3}	2.496	2.404	224	0.96

Table 3.1

Solution of the Torsion Problem by the Five-Point Formula

Mesh Size $h=\frac{1}{n}$	Number of Interior Gridpoints	Max. Trun. Error in Nine-Point ormula	Max. Error in Solution of System	Time in Secs by Gauss Elim. t_{G9}	Solution by Point-Gauss-Seidel		$\frac{t_{S9}}{t_{G9}}$
					Time in Secs t_{S9}	Number of Iterations	
$\frac{1}{4}$	16	7.2×10^{-3}	1.0×10^{-6}	0.008	0.046	69	5.75
$\frac{1}{5}$	25	3.1×10^{-3}	5.9×10^{-6}	0.017	0.097	91	5.70
$\frac{1}{10}$	100	2.3×10^{-3}	1.3×10^{-4}	0.192	0.960	214	5.00
$\frac{1}{20}$	400	1.5×10^{-3}	5.7×10^{-3}	2.614	1.662	94	0.64

Table 3.2

Solution of the Torsion Problem by the Nine-Point Formula

then comparing the results with those obtained by the Gaussian elimination method. The time required to obtain the same accuracy by both methods can then be found. The timing results shown in Tables 3.1 and 3.2 exclude the time required for the input and output of data, and are accurate to within 2 percent.

3.4.2 Interpretation of Results When solving a problem in partial differential equations, a finite-difference formula with a suitable mesh size is chosen so that the finite-difference approximation is of sufficient accuracy. The resulting system of equations need then be solved to the accuracy that the finite-difference approximation warrants. This feature is easily incorporated into iterative methods by terminating the procedure when sufficient accuracy is attained. For direct methods, however, the accuracy is entirely determined by round-off errors, so that the accuracy of the solution of a system of equations does not usually correspond with the accuracy of the finite-difference approximation. Therefore, the fact that the Gaussian elimination method is much faster than the point-Gauss-Seidel method for mesh sizes of $\frac{1}{4}$, $\frac{1}{5}$, and $\frac{1}{10}$ in Tables 3.1 and 3.2 does not indicate that the method of Gaussian elimination should be preferred over the point-Gauss-Seidel method, since the systems are solved by the latter method to an unnecessarily

high degree of accuracy.

If the point-Gauss-Seidel method is used to solve the system of equations to the same accuracy as that of the finite-difference approximation (see Table 3.3), the time required is decreased for nearly all cases. For case (*) in Figure 3.3 (where Laplace's equation is approximated by the nine-point formula with a mesh size of $\frac{1}{20}$), the time is increased, because it is necessary to improve the accuracy of the solution of the system to that of the nine-point formula.

Ignoring case (*) for the moment, we see from the table entries, t^*_{S5}/t_{G5} and t^*_{S9}/t_{G9} in Table 3.3, that the Gaussian elimination method is still faster in nearly all cases (excluding case † , where the five-point formula with a mesh size of $\frac{1}{20}$ is used) than the point-Gauss-Seidel method. For case (*) , the final solution of the problem obtained by the Gaussian elimination method (with time t_{G9}), due to round-off errors, is less accurate than that obtained by the point-Gauss-Seidel method (with time t^*_{S9}). The solution of the system obtained by the Gaussian elimination method for this case might, however, be sufficiently improved by applying an iterative improvement to the solution (see Ralston [16]) with only a moderate increase of time (approximately $2n^3$ additional multiplications and $2(n+1)^3$ additions are required for

Mesh Size	Five-Point Formula				Nine-Point Formula			
	Error in Solution u	Time in Secs t^*_{S5}	t^*_{S5}/t_{G5}	Number of Iterations	Error in Solution u	Time in Secs t^*_{S9}	t^*_{S9}/t_{G9}	Number of Iterations
$\frac{1}{4}$	1.7×10^{-2}	0.010	1.43	22	7.2×10^{-3}	0.016	2.00	24
$\frac{1}{5}$	8.9×10^{-3}	0.025	1.67	35	3.1×10^{-3}	0.043	2.53	40
$\frac{1}{10}$	5.4×10^{-3}	0.261	1.56	96	2.3×10^{-3}	0.511	2.68	115
$\frac{1}{20}$	3.7×10^{-3}	2.108	$0.84^{\dagger}$	197	1.5×10^{-3}	5.612	2.25*	316

Table 3.3

Solution by the Point-Gauss-Seidel Method

one iteration improvement); so that, the Gaussian elimination method should still be faster than the point-Gauss-Seidel method in obtaining the same accuracy to the final solution of the problem.

For this investigation, we have chosen an initial guess to the solution of the system always to be zero before applying the point-Gauss-Seidel iteration. For most problems, however, a better estimate is known (for example, a better estimate to u for Problem 3.1 may be that $u = x^2 + y^2$ in R^*), so that the time required for the point-Gauss-Seidel method is once again decreased. We therefore conclude (with uncertainty since an estimate to the solution is subject to human discretion) that for all cases, the point-Gauss-Seidel method may be just as efficient as the method of Gaussian elimination in obtaining the same accuracy to the final solution of the problem. For case (†) , the point-Gauss-Seidel method is undoubtedly superior.

CHAPTER IV

FLOW OF A PERFECT FLUID IN AN L-SHAPED CHANNEL

4.1 Statement of the Problem

We consider the flow of an incompressible, perfect fluid through a two-dimensional channel with parallel sides in which there is a right-angle bend (see Collatz [4]), or Thom and Apelt [20]). It is assumed that the fluid flows in with unit velocity across the entrance AF and out with unit velocity across the exit CD , where ABC and DEF are the walls of the channel as in Figure 4.1.

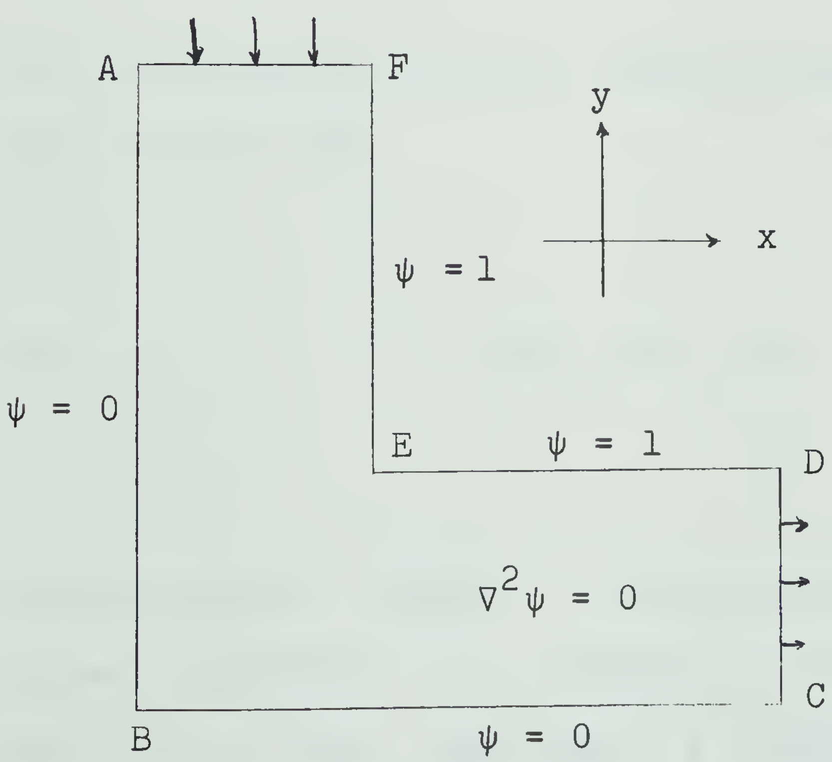

Figure 4.1 - L-Shaped Channel

The velocity components of the fluid v_x and v_y can be written as the partial derivatives of the stream function ψ , or of the potential function ϕ :

(4.1.1) $$v_x = \frac{\partial \phi}{\partial x} = \frac{\partial \psi}{\partial y} ,$$

and

(4.1.2) $$v_y = \frac{\partial \phi}{\partial y} = \frac{\partial \psi}{\partial x} .$$

It then follows from the Cauchy-Riemann equations (4.1.1) and (4.1.2) that

(4.1.3) $$\nabla^2 \phi = \nabla^2 \psi = 0$$

in the region ABCDEF . Furthermore, we have that the stream function ψ is constant on the boundary ABC , as well as on DEF ; and that, ψ varies linearly on AF and CD if the two limbs of the channel extend sufficiently far to infinity.

The problem in terms of ψ then becomes the following Dirichlet problem:

Problem 4.1 *Find* ψ *so that*

$$\nabla^2\psi = 0 \text{ in the region } ABCDEF ,$$

subject to the boundary conditions

$$(4.1.4) \qquad \psi = \begin{cases} 0 \text{ along } ABC \\ 1 \text{ along } DEF \\ \text{varies linearly along } AF \text{ and } CD . \end{cases}$$

If the distance from A to B is chosen to be the same as the distance from B to C , the problem becomes symmetric about the line BE and it is necessary to consider only one limb of the channel, ABEF , say. An analytical solution to the problem has not yet been found, so that numerical methods must be resorted to.

For solving the problem by finite-difference methods, the limb ABEF is first superimposed by a square grid-system of mesh length h as shown in Figure 4.2.

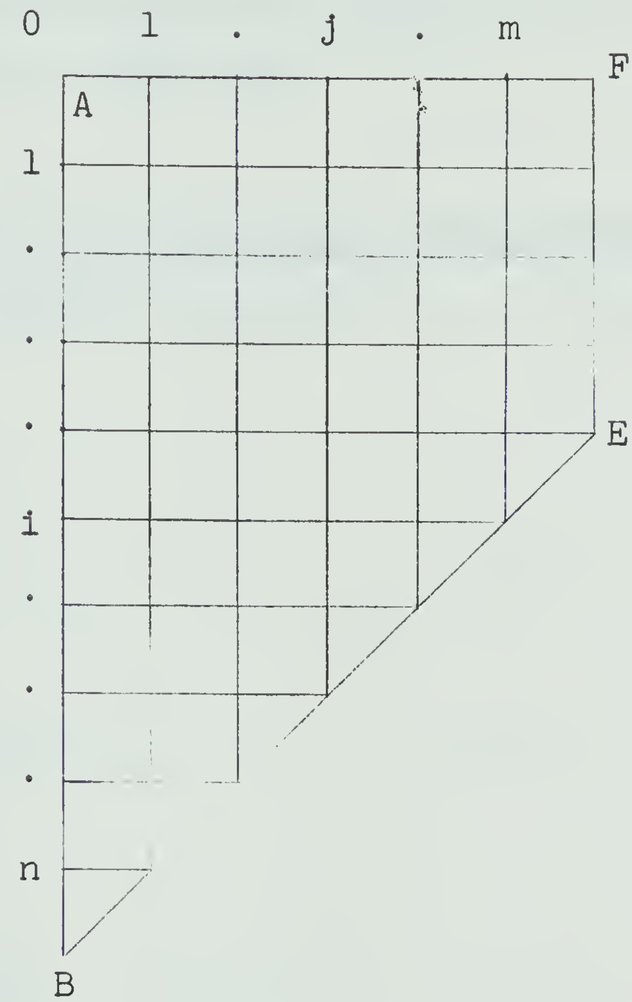

Figure 4.2 Grid-System for the L-Shaped Channel

Suppose that m mesh lines are required in the x-direction and n in the y-direction. Finite-difference formulae are then used to approximate Laplace's equation at the interior gridpoints

$$(i,j)\ ,\quad i=1,2,\ldots,n-j+1\ ,\qquad j=1,2,\ldots,m\ ; \tag{4.1.5}$$

and, the problem is reduced to that of solving a linear algebraic system of equations.

4.2 Solution by Use of the Five-Point Formula

From the boundary condition (4.1.4), we have that

$$\psi(i,0) = 0 , \qquad i=1,2,\ldots,n+1 ,$$

$$\psi(i,m+1) = 1 , \qquad i=1,2,\ldots,n-m , \tag{4.2.1}$$

$$\psi(0,j) = -j/(m+1) , \qquad j=1,2,\ldots,m .$$

For interior gridpoints, the five-point formula (4.2.3) for Laplace's equation yields

$$\psi(i,j) = \frac{1}{4}\,[\psi(i,j+1) + \psi(i-1,j) + \psi(i,j-1) + \psi(i+1,j)] ,$$

$$i=1,2,\ldots,n-j , \tag{4.2.2}$$

$$j=1,2,\ldots,m ;$$

and, from the symmetry of the problem,

$$\psi(n-j+1) = \frac{1}{2}\,[\psi(n-j,j) + \psi(n-j+1,j-1)] \;, \qquad j=1,2,\ldots,m \;. \tag{4.2.3}$$

The system (4.2.2) and (4.2.3) can be solved by the point-Gauss-Seidel method by recursively applying formula (4.2.2) and (4.2.3) at the gridpoints (i,j) until sufficient accuracy is attained. An initial guess to the solution ψ at the interior gridpoints must first, however, be made, and a consistent ordering of points chosen. One such ordering is given by (4.1.5). The point-Gauss-Seidel method requires $m(n+1) - m(m+1)/2$ multiplications and $4mn - m(2m+1)$ additions for each iteration; that is, for each application of (4.2.2) or (4.2.3) to the points (4.1.5).

The system can also be solved directly by the method of Gaussian elimination. Let L_k denote the matrix of dimension $(n-k) \times (n-k+1)$,

(4.2.4) $$L_k = \begin{bmatrix} 1 & & & & & \\ & 1 & & & & \\ & & \cdot & & & \\ & & & \cdot & & \\ & & & & \cdot & \\ & & & & 1 & \\ & & & & & 2 & 0 \end{bmatrix} ,$$

where the elements which are not assigned values are considered to be zero; I_k , the matrix of dimension $(n-k+1) \times (n-k)$,

(4.2.5) $$I_k = \begin{bmatrix} 1 & & & & & \\ & 1 & & & & \\ & & \cdot & & & \\ & & & \cdot & & \\ & & & & \cdot & \\ & & & & & 1 \\ & & & & & 0 \end{bmatrix} ;$$

and D_k , the tridiagonal matrix of order $n-k+1$,

$$(4.2.6) \qquad D_k = \begin{bmatrix} -4 & 1 & & & & & \\ 1 & -4 & 1 & & & & \\ & 1 & -4 & 1 & & & \\ & & & \cdot & & & \\ & & & & \cdot & & \\ & & & & & \cdot & \\ & & & & 1 & -4 & 1 \\ & & & & & 2 & -4 \end{bmatrix} .$$

If B_k and Ψ_k are column vectors of dimension $n-k+1$ defined by

$$(4.2.7) \qquad \begin{aligned} B_k &= [-k/(m+1),0,0,\ldots,0]^T , \quad k=1,2,\ldots,m-1 , \\ B_m &= [-1,-1,\ldots,-1]^T , \end{aligned}$$

and

$$(4.2.8) \qquad \Psi_k = [\psi(1,k),\psi(2,k),\ldots,\psi(n-k+1,k)]^T ;$$

then, using the same ordering of points (see (3.1.5)) as for the point-Gauss-Seidel method, the system (4.2.2) and (4.2.3) can be written as

$$A\Psi = B \,, \tag{4.2.9}$$

where A is the block-tridiagonal matrix of order $mn - m(m-1)/2$,

$$A = \begin{bmatrix} D_1 & I_1 & & & & & \\ L_1 & D_2 & I_2 & & & & \\ & L_2 & D_3 & I_3 & & & \\ & & & \ddots & & & \\ & & & L_{m-2} & D_{m-1} & I_{m-1} \\ & & & & L_{m-1} & D_m \end{bmatrix} , \tag{4.2.10}$$

and Ψ and B are the block vectors,

$$\Psi = [\Psi_1^T, \Psi_2^T, \ldots, \Psi_m^T]^T \,, \tag{4.2.11}$$

and

$$B = [B_1^T, B_2^T, \ldots, B_m^T]^T \,. \tag{4.2.12}$$

It can be seen from (4.2.10) that decomposition of A by Gaussian elimination can be performed by blocks. That is, decomposition of D_1 has effect on I_1, L_1 and D_2 only (denoted the new D_2 by D_2^*), of D_2^* on I_2, L_2 and D_3 only (denoted the new D_3 by D_3^*), and of D_k^* on I_k, L_k and D_{k+1} only (denote the new D_{k+1} by D_{k+1}^*).

For example, when decomposing the k'th block for $n-k+1 = 6$, elimination is performed on

(4.2.13)
$$\left[\begin{array}{c|c} D_k^* & L_k \\ \hline L_k & D_{k+1} \end{array}\right] = \left[\begin{array}{cccccc|cccccc}
x & x & x & x & x & x & 1 & & & & & \\
x & x & x & x & x & x & & 1 & & & & \\
x & x & x & x & x & x & & & 1 & & & \\
x & x & x & x & x & x & & & & 1 & & \\
x & x & x & x & x & x & & & & & 1 & \\
x & x & x & x & x & x & & & & & 0 & \\
\hline
1 & & & & & & -4 & 1 & & & & \\
& 1 & & & & & 1 & -4 & 1 & & & \\
& & 1 & & & & & 1 & -4 & 1 & & \\
& & & 1 & & & & & 1 & -4 & 1 & \\
& & & & 2 & 0 & & & & 2 & -4 &
\end{array}\right],$$

where x denotes an arbitrary value. At the third stage of the elimination of (4.2.13), we have

$$\left[\begin{array}{cccccc|ccccc}
x & x & x & x & x & x & 1 & & & & \\
 & x & x & x & x & x & x & 1 & & & \\
 & & x & x & x & x & x & x & 1 & & \\
 & & x & x & x & x & x & x & & 1 & \\
 & & x & x & x & x & x & x & & & 1 \\
 & & x & x & x & x & x & x & & & 0 \\
\hline
 & & x & x & x & x & x & x & & & \\
 & & x & x & x & x & x & x & & & \\
 & & 1 & & & & & 1 & -4 & & \\
 & & & 1 & & & & & 1 & -4 & 1 \\
 & & & & 2 & & & & & 2 & -4
\end{array}\right],$$

Figure 4.3

and, after the final elimination, we have

$$\left[\begin{array}{cccccc|ccccc}
x & x & x & x & x & x & 1 & & & & \\
 & x & x & x & x & x & x & 1 & & & \\
 & & x & x & x & x & x & x & 1 & & \\
 & & & x & x & x & x & x & x & 1 & \\
 & & & & x & x & x & x & x & x & 1 \\
 & & & & & x & x & x & x & x & x \\
\hline
 & & & & & & x & x & x & x & x \\
 & & & & & & x & x & x & x & x \\
 & & & & & & x & x & x & x & x \\
 & & & & & & x & x & x & x & x \\
 & & & & & & x & x & x & x & x
\end{array}\right],$$

Figure 4.4

where the lower right-hand block in Figure 4.4 is D^*_{k+1} . We have assumed in Figures 4.3 and 4.4, as for the Torsion Problem 3.1, that no pivoting was performed during the elimination process.

Since the decomposed matrix A is of band form, with band width of at most n (assuming that pivoting is not used and the 1's on top of the band are ignored (see Figure 4.4)), a matrix of dimension $[mn - m(m-1)/2) \times n$ is large enough to store all the results of the forward elimination of A . Approximately $(m-1)n^3$ multiplications and the same number of additions are required for the forward-elimination and back-substitution processes.

4.3 Solution by Use of the Nine-Point Formula

When the nine-point formula (2.2.16) is used to approximate Laplace's equation at the interior gridpoints (4.1.5), the problem is reduced to solving the following system of linear algebraic equations:

$$\psi(i,j) = \frac{1}{20}\{4[\psi(i,j+1) + \psi(i-1,j) + \psi(i,j-1) + \psi(i+1,j)]$$

$$+ \psi(i-1,j+1) + \psi(i-1,j-1) + \psi(i+1,j-1) + \psi(i+1,j+1)\},$$

$$(4.3.1) \qquad i=1,2,\ldots,n-j-1 \ ,$$

$$j=1,2,\ldots,m \ ;$$

and, from the symmetry of the problem (see Figure 4.2),

$$\psi(n-j,j) = \frac{1}{19}\{4[\psi(n-j,j+1) + \psi(n-j-1,j) + \psi(n-j,j-1)$$

$$+ \psi(n-j+1,j)] + \psi(n-j-1,j+1)$$

$$+ \psi(n-j,j-1) + \psi(n-j+1,j-1)\} ,$$

(4.3.2) $\qquad j=1,2,\ldots,m ;$

$$\psi(n-j+1,j) = \frac{1}{20}\{8[\psi(n-j,j) + \psi(n-j+1,j-1)]$$

$$+ \psi(n-j,j+1) + 2\psi(n-j,j-1) + \psi(n-j+2,j-1)\} ,$$

$$j=1,2,\ldots,m ;$$

given that, along the boundary, the boundary condition (4.2.1) holds true.

Let L_k be the matrix of dimension $(n-k) \times (n-k+1)$,

$$(4.3.3) \qquad L_k = \begin{bmatrix} 4 & 1 & & & & & & & \\ 1 & 4 & 1 & & & & & & \\ & 1 & 4 & 1 & & & & & \\ & & & \cdot & & & & & \\ & & & & \cdot & & & & \\ & & & & & \cdot & & & \\ & & & & & 1 & 4 & 1 & \\ & & & & & & 2 & 8 & 1 \end{bmatrix} ;$$

I_k , the matrix of dimension $(n-k+1) \times (n-k)$,

$$(4.3.4) \qquad I_k = \begin{bmatrix} 4 & 1 & & & & & & \\ 1 & 4 & 1 & & & & & \\ & 1 & 4 & 1 & & & & \\ & & & \cdot & & & & \\ & & & & \cdot & & & \\ & & & & & \cdot & & \\ & & & & & 1 & 4 & 1 \\ & & & & & & 1 & 4 \\ & & & & & & & 1 \end{bmatrix} ;$$

D_k , the tridiagonal matrix of order $n-k+1$,

$$(4.3.5)\quad D_k = \begin{bmatrix} -20 & 4 & & & & & & \\ 4 & -20 & 4 & & & & & \\ & 4 & -20 & 4 & & & & \\ & & & \cdot & & & & \\ & & & & \cdot & & & \\ & & & & & \cdot & & \\ & & & & 4 & -20 & 4 & \\ & & & & & 4 & -19 & 4 \\ & & & & & & 8 & -20 \end{bmatrix} ;$$

B_k , the column vector of dimension $n-k+1$,

$$B_k = [-6k/(m+1),0,0,\ldots,0]^T , \quad k=1,2,\ldots,m-1 ,$$

(4.3.6)

$$B_m = [-(5m+11)/(m+1),-6,-6,\ldots,-6]^T ;$$

and Ψ_k , as defined in (4.2.8). Then the system (4.3.1) and (4.3.2) can be written in matrix form as in (4.2.9).

As for the five-point formula, decomposition of A by the Gaussian-elimination method can be performed in blocks; so that, for the k'th block (suppose $n-k+1 = 6$), elimination is performed on

$$(4.3.7)\qquad \left[\begin{array}{c|c} D_k^* & L_k \\ \hline L_k & D_{k+1} \end{array}\right] = \left[\begin{array}{cccccc|ccccc}
x & x & x & x & x & x & 4 & 1 & & & \\
x & x & x & x & x & x & 1 & 4 & 1 & & \\
x & x & x & x & x & x & & 1 & 4 & 1 & \\
x & x & x & x & x & x & & & 1 & 4 & 1 \\
x & x & x & x & x & x & & & & 1 & 4 \\
x & x & x & x & x & x & & & & & 1 \\
\hline
4 & 1 & & & & & -20 & 4 & & & \\
1 & 4 & 1 & & & & 4 & -20 & 4 & & \\
 & 1 & 4 & 1 & & & & 4 & -20 & 4 & \\
 & & 1 & 4 & 1 & & & & 4 & -19 & 4 \\
 & & & 2 & 8 & 1 & & & & 8 & -20
\end{array}\right].$$

At the third stage of the elimination, we have

$$\left[\begin{array}{cccccc|ccccc}
x & x & x & x & x & x & 4 & 1 & & & \\
 & x & x & x & x & x & x & x & 1 & & \\
 & & x & x & x & x & x & x & x & 1 & \\
 & & x & x & x & x & x & x & x & 4 & 1 \\
 & & x & x & x & x & x & x & x & 1 & 4 \\
 & & x & x & x & x & x & x & x & & 1 \\
\hline
 & & x & x & x & x & x & x & x & & \\
 & & x & x & x & x & x & x & x & & \\
 & & x & x & x & x & x & x & x & 4 & \\
 & & 1 & 4 & 1 & & & & 4 & -19 & 4 \\
 & & & 2 & 8 & 1 & & & & 8 & -20
\end{array}\right];$$

Figure 4.5

and, after the final stage of the elimination, we have

$$
\left[\begin{array}{cccccc|ccccc}
x & x & x & x & x & x & 4 & 1 & & & \\
 & x & x & x & x & x & x & x & 1 & & \\
 & & x & x & x & x & x & x & x & 1 & \\
 & & & x & x & x & x & x & x & x & 1 \\
 & & & & x & x & x & x & x & x & x \\
 & & & & & x & x & x & x & x & x \\
\hline
 & & & & & & x & x & x & x & x \\
 & & & & & & x & x & x & x & x \\
 & & & & & & x & x & x & x & x \\
 & & & & & & x & x & x & x & x \\
 & & & & & & x & x & x & x & x
\end{array}\right] .
$$

Figure 4.6

A matrix of dimension $[mn - m(m-1)/2] \times [n+1]$ is necessary to store the results of the forward-elimination of A for the nine-point formula. Approximately $(m-1)(n+1)^3$ multiplications, as well as additions, are required for the complete Gaussian elimination process; as compared with the point-Gauss-Seidel method which requires $2mn - m(m-1)$ multiplications and $7mn - \frac{1}{2}m(m+1)$ additions for each iteration.

4.4 Numerical Results

The same procedure for the presentation of results as in Section 3.4 for the torsion problem is used in this section, so that most of the observations that are made there are also true in this section. Attention is thus focused, not on the accuracy of the finite-difference approximation, but on the accuracy with which the resulting system of equations is solved. Fortran IV programs to solve the Problem 4.1 can be found in Appendix IV.

4.4.1 Test Data and Results An example of the problem is given in Thom and Apelt [20] where the width of the channel AF is 1 unit and the length of the outside wall of the channel AB is 2.75 units (see Figure 4.2). For this example, the five-point formula (2.2.3) and the nine-point formula (2.2.16) are used to approximate Laplace's equation, and the resulting systems of equations are solved by both the point-Gauss-Seidel and the Gaussian elimination methods. For the point-Gauss-Seidel method, an initial estimate of the solution is always zero.

A summary of the results for mesh sizes of $\frac{1}{4}$, $\frac{1}{8}$, and $\frac{1}{16}$ is presented in Tables 4.1 and 4.2, where the numerical values are obtained as in Section 3.4.1.

4.4.2 Interpretation of Results The finite-difference approximations to Laplace's equation are not as good as might

Mesh Size h	m	n	Number of Interior Gridpoints	Max. Trun. Error in Five-Point Formula	Max. Error in Solution of System	Time in in Secs by Gauss. Elim. t_{G5}	Solution by Point-Gauss-Seidel		$\frac{t_{S5}}{t_{G5}}$
							Time in Secs t_{S5}	Number of Iterations	
$\frac{1}{4}$	3	10	27	2.8×10^{-2}	6.2×10^{-6}	0.026	0.031	34	1.20
$\frac{1}{8}$	7	21	126	1.4×10^{-2}	3.2×10^{-5}	0.623	0.675	99	1.08
$\frac{1}{16}$	15	43	540	4.8×10^{-3}	1.7×10^{-4}	10.442	6.827	232	0.66

Table 4.1

Five-Point Formula for the L-Shaped Channel

Mesh Size h	m	n	Number of Interior Gridpoints	Max. Trun. Error in Five-Point Formula	Max. Error in Solution of System	Time in in Secs by Gauss. Elim. t_{G9}	Solution by Point-Gauss-Seidel		$\frac{t_{S9}}{t_{G9}}$
							Time in Secs t_{S9}	Number of Iterations	
$\frac{1}{4}$	3	10	27	1.5×10^{-2}	1.3×10^{-5}	0.040	0.048	32	1.20
$\frac{1}{8}$	7	21	126	9.2×10^{-3}	6.4×10^{-5}	0.681	0.716	98	1.05
$\frac{1}{16}$	15	43	540	2.7×10^{-3}	3.5×10^{-4}	10.957	7.792	256	0.71

Table 4.2

Five-Point Formula for the L-Shaped Channel

at first be anticipated when using formulae with truncation errors of $O(h^4)$ and $O(h^6)$; however, due to the singularity at the point C (Figure 4.1), the higher-order terms, which are truncated in the Taylor's series expansion, are no longer insignificant. To compensate for this fact, the size of the mesh might be decreased near this point. Singularities for this investigation are, however, ignored.

For mesh sizes of $\frac{1}{4}$ and $\frac{1}{8}$, Tables 4.1 and 4.2 show that the speeds of the point-Gauss-Seidel and the Gaussian elimination methods are approximately the same; while for a mesh size $\frac{1}{16}$, the point Gauss-Seidel method is considerably faster. The time required to obtain the solutions by the point-Gauss-Seidel method can, however, in all cases be decreased by choosing a more suitable initial guess to the solution (Ψ varies linearly in Definition 4.1, for example). Furthermore, the solution of the system need only be as accurate as the finite-difference approximation, so that the number of point-Gauss-Seidel iterations (and thus computing time) can be reduced accordingly. The user has no such control on the accuracy, or the time when using direct methods in solving systems of equations.

We can thus safely conclude that for the problem of the L-shaped channel (Definition 4.1), the point-Gauss-Seidel method is superior to the Gaussian elimination method.

CHAPTER V

A FREE-BOUNDARY PROBLEM

5.1 Statement of the Problem

We consider a simplified problem of fresh-water flow in a confined coastal aquifer (see Charmonman [3]). Fresh water is supplied by a series of parallel canals with intermediate drains (Figure 5.1), where the fresh and salt water interface is not initially known (thus, a free-boundary problem). The configuration is symmetric about the centre line of the canal, so that only half of it, as in Figure 5.2, need be considered.

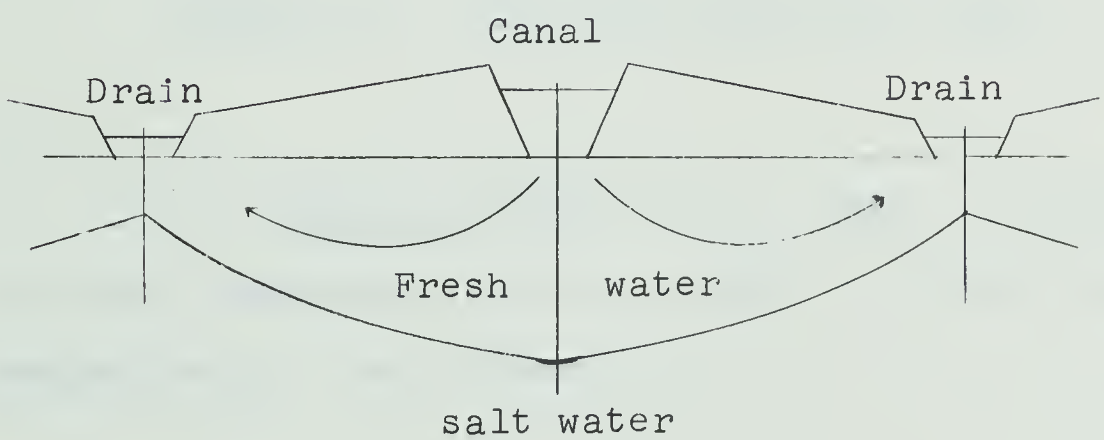

Figure 5.1 Physical Plane

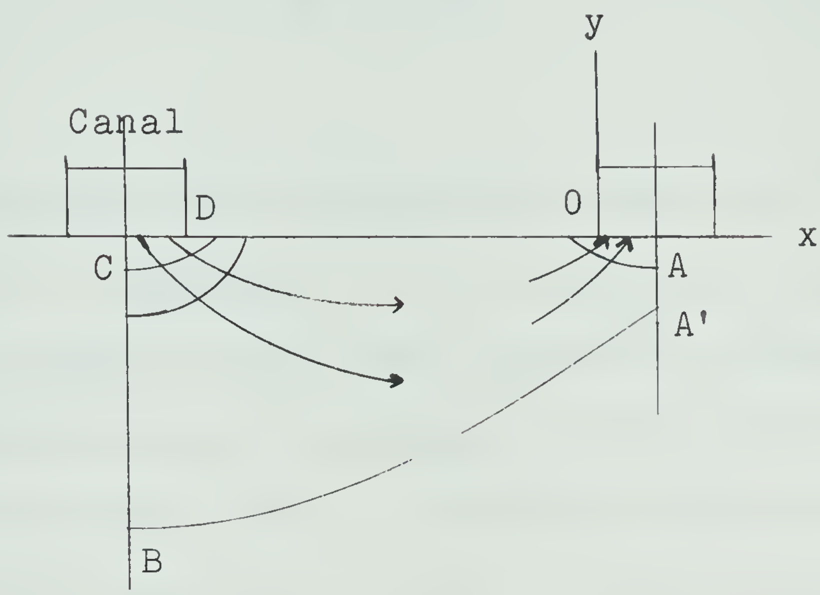

Figure 5.2 Simplified Physical Plane

5.1.1 Physical Plane If u_x and v_y represent the velocity components in the x and y directions, respectively, then we have

$$u_x = \frac{\partial \phi}{\partial x} = \frac{\partial \psi}{\partial y}, \tag{5.1.1}$$

and

$$v_y = \frac{\partial \phi}{\partial y} = -\frac{\partial \psi}{\partial x} , \tag{5.1.2}$$

where ϕ represents the potential function and ψ the stream function. Equations (5.1.1) and (5.1.2) are sometimes known as Darcy's law, which holds true under a few restrictive conditions on the medium of flow and on the aquifer (see Charmonman [2]). Differentiating (5.1.1) and (5.1.2) with respect to x and y, we see that ψ and ϕ satisfy Laplace's equation in the region of flow; that is,

$$\frac{\partial^2 \psi}{\partial x^2} + \frac{\partial^2 \psi}{\partial y^2} = 0 , \tag{5.1.3}$$

and

$$\frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} = 0 . \tag{5.1.4}$$

The boundary conditions for the problem are derived in [3] and are given in Table 5.1 in dimensionless form. The problem is therefore a mixed boundary-value problem and is formally described by

[illegible]

$$[illegible]$$

[illegible]

$$[illegible]$$

[illegible]

Boundary Segment	Boundary Condition with Respect to ψ	Boundary Condition with Respect to ϕ
OA	$\frac{\partial\psi}{\partial y} = 0$	$\phi = 0$
AA'	$\psi = 0$	$\frac{\partial\phi}{\partial x} = 0$
A'B	$\psi = 0$	$\phi = y$
BC	$\psi = 0$	$\frac{\partial\phi}{\partial x} = 0$
CD	$\frac{\partial\psi}{\partial y} = 0$	$\phi =$ constant ϕ_c
DO	$\psi = 1$	$\frac{\partial\phi}{\partial y} = 0$

Table 5.1

Boundary Conditions in the Physical Plane

Problem 5.1 *Find* ψ *and* ϕ *in the region* R *of fresh-water flow (Figure 5.2) so that Laplace's equations (5.1.3) and (5.1.4), subject to the boundary conditions given in Table 5.1, are satisfied; and, determine the position of the free boundary* $A'B$ *so that the Cauchy-Riemann equations (5.1.1) and (5.1.2) hold true for all points in the region* R .

Since an analytical solution has not yet been found, numerical methods must be used to solve the problem. The greatest difficulty arises in finding the position of the free boundary $A'B$, which is determined by a trial-and-error method.

5.1.2 Complex-Potential Plane Obtaining the position of the free-boundary is greatly simplified by transforming the problem into the complex-potential plane, although an analytical solution to the transformed problem still cannot be found.

If we let

$$w = \phi + i\psi \,, \tag{5.1.5}$$

then w is an analytic function in the region R of the z-plane, where

$$z = x + iy \,, \tag{5.1.6}$$

inasmuch as the Cauchy-Riemann equations hold true and $\phi(x,y)$ and $\psi(x,y)$ both satisfy Laplace's equation in R. Thus, provided that $\frac{dw}{dz}$ exists and is different from zero in R, z can be written as

(5.1.7) $$x + iy = f(\phi + i\psi) ,$$

and is analytic in the region R^* of the w-plane (the complex-potential plane), where R^* is defined by $R = f(R^*)$. Therefore, the inverse Cauchy-Riemann equations

(5.1.8) $$\frac{\partial x}{\partial \phi} = \frac{\partial y}{\partial \psi} ,$$

and

(5.1.9) $$\frac{\partial x}{\partial \psi} = -\frac{\partial y}{\partial \phi}$$

are true in R^* , and the inverse Laplace's equations

(5.1.10) $$\frac{\partial^2 x}{\partial \phi 2} + \frac{\partial^2 x}{\partial \psi 2} = 0 ,$$

and

(5.1.11) $$\frac{\partial^2 y}{\partial \phi^2} + \frac{\partial^2 y}{\partial \psi^2} = 0$$

are satisfied.

Let f^{-1} map the region R of the physical plane (Figure 5.2) into the complex-potential plane so that R^* $(f^{-1}(R) = R^*)$ is the rectangular region shown in Figure 5.3.

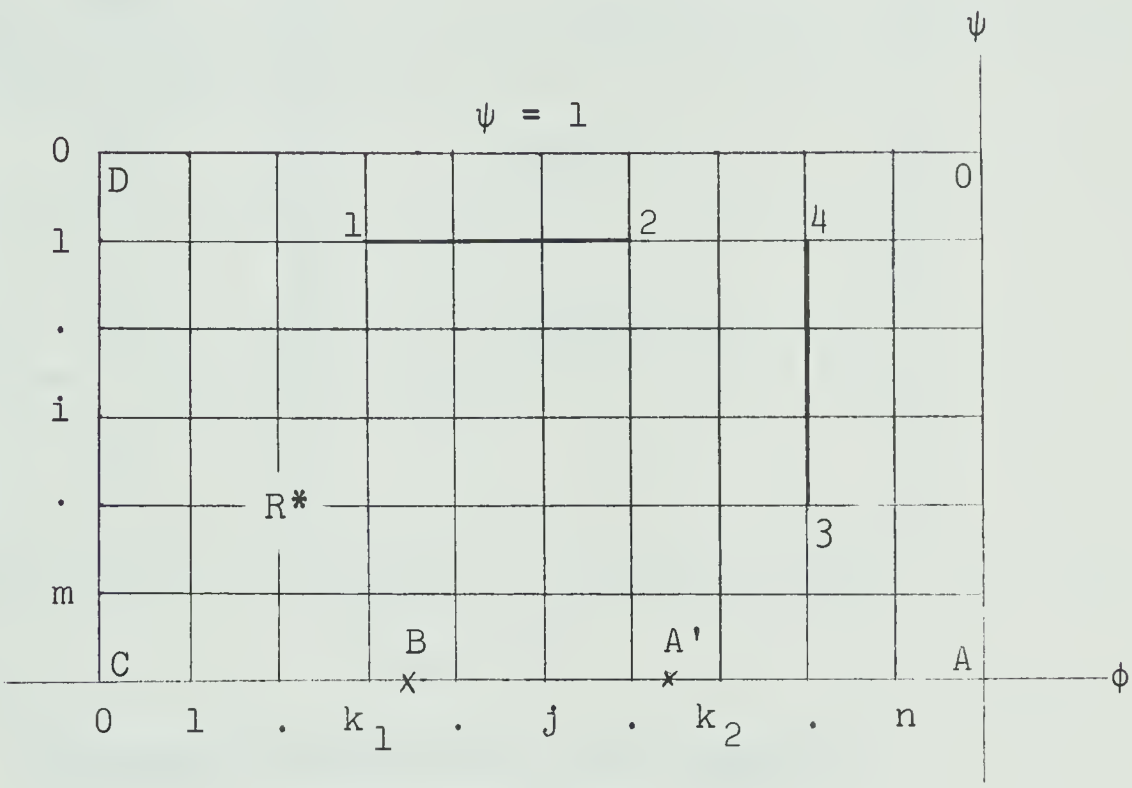

Figure 5.3 Complex-Potential Plane

The boundary conditions with respect to $y(\phi,\psi)$ along the boundary of R^* can then immediately be obtained from Table 5.1 and are summarized in Table 5.2.

Boundary Segment	Boundary Condition with Respect to y
OA	$y = 0$
AA'	$\partial y/\partial\psi = 0$
A'B	$y = \phi$
BC	$\partial y/\partial\psi = 0$
CD	$y = 0$
DO	$y = 0$

Table 5.2

Boundary Conditions in the Complex-Potential Plane

The boundary conditions with respect to $x(\phi,\psi)$ along all segments of the boundary of R^* are, however, not available. The x-values must thus be found by integrating the inverse Cauchy-Riemann equations (5.1.8) and (5.1.9).

Along lines of constant ψ , such as the vertical line from point 1 to the point 2 in Figure 5.3, we have

$$(5.1.12) \qquad x_2 - x_1 = \int_{\phi_1}^{\phi_2} \frac{\partial x}{\partial \phi} d\phi + g\,(\psi) \Big|_{\psi_1}^{\psi_2} ,$$

which, on substitution of the inverse Cauchy-Riemann equation (5.1.8), becomes

$$(5.1.13) \qquad x_2 - x_1 = \int_{\phi_1}^{\phi_2} \frac{\partial y}{\partial \psi} d\phi + g(\psi) \Big|_{\psi_1}^{\psi_2} .$$

But along lines of constant ψ , $g(\psi_1) = g(\psi_2)$, so that (5.1.13) reduces to

$$(5.1.14) \qquad x_2 - x_1 = \int_{\phi_1}^{\phi_2} \frac{\partial y}{\partial \psi} d\phi .$$

Similarly, along lines of constant ϕ , such as the horizontal line from point 3 to point 4 in Figure 5.3, we have, from the inverse Cauchy-Riemann equation (5.1.9), that

(5.1.15) $$x_4 - x_3 = -\int_{\psi_3}^{\psi_4} \frac{\partial y}{\partial \phi}\, d\psi \ .$$

Furthermore, x is constant on AA' and BC ; that is,

(5.1.16) $$x(P) = x(A) \ , \quad P \text{ on } AA' \ ,$$

and

(5.1.17) $$x(P) = x(C) \ , \quad P \text{ on } BC \ .$$

The mixed boundary-value problem in the complex-potential plane can then be described by

<u>Problem 5.2</u>. *Find $y = y(\phi, \psi)$, harmonic in the region R^* (Figure 5.3) with the boundary conditions, as given in Table 5.2, satisfied; and determine $x = (\phi, \psi)$ in R^* by use of the integrals (5.1.14) and (5.1.15), where the position of the points A' and B is chosen so that (5.1.16) and (5.1.17) hold true.*

Although the trial-and-error methods is used to find

the correct position of the points A' and B , only two points (not an infinity of points as in the physical plane) need be considered.

5.2 Solution in the Physical Plane

In this section only, because of the difficulty in finding the free boundary, we further simplify the problem by considering the flow of fresh water from a canal without intermediate drains. This is a special case of Problem 5.1 in that the segment AA' of the boundary of the region R is entirely compressed (see Figure 5.4). The boundary conditions along the remaining segments of the boundary, where OA now represents the horizontal outflow face, hold as in Table 5.1.

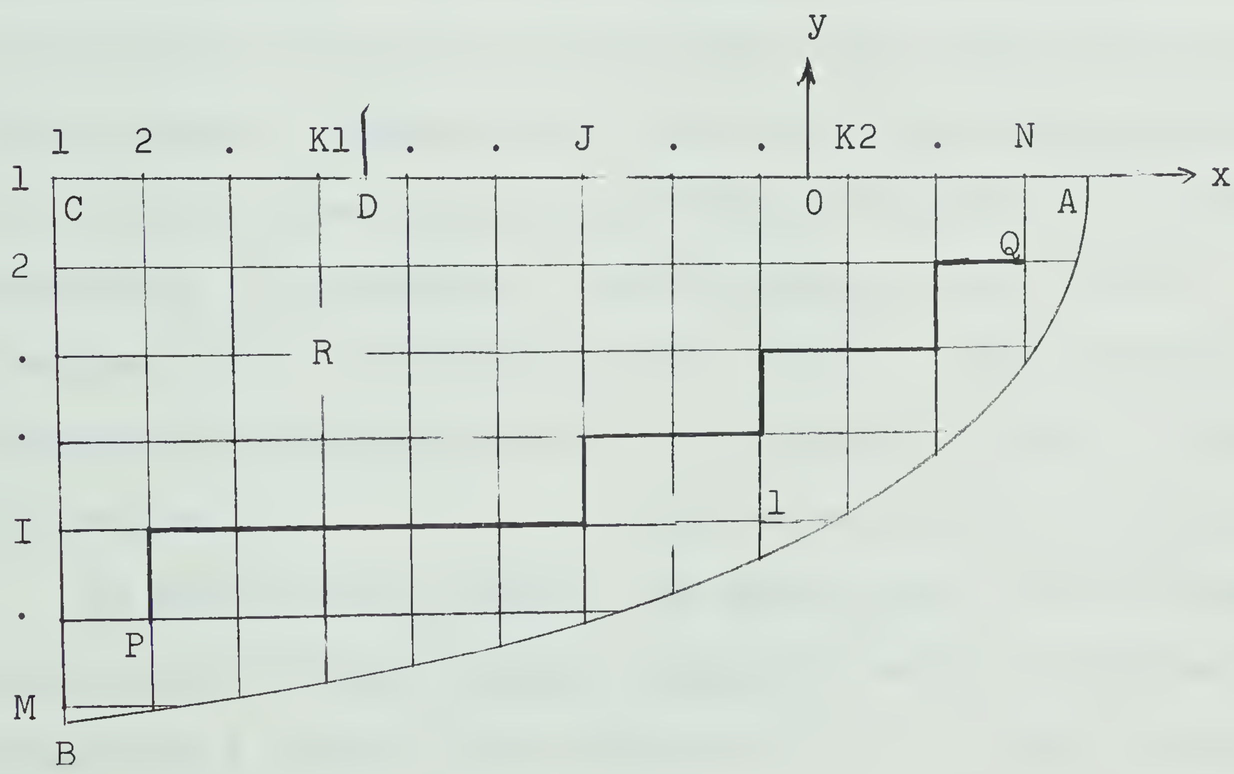

Figure 5.4 Grid System in the Physical Plane

5.2.1 Method of Solution In solving the problem numerically, an initial guess as to the shape and position of the free boundary AB is made. $\phi(x,y)$ and $\psi(x,y)$ are then indpendently determined in the region R of fresh-water flow by the method of finite-differences, and the Cauchy-Riemann equations tested to see if they are satisfied. If they are, $w = \phi + i\psi$ is an analytic function in R and the problem is solved; if not, the free boundary is adjusted and the procedure repeated.

In using the method of finite-differences to determine ϕ and ψ, the region R is first superimposed by a square grid-system as in Figure 5.4. Laplace's equation is then approximated by: the five-point formula (2.2.3) at all interior gridpoints of R, except for those near the free boundary AB ; the irregular-star formulae, (2.2.4), (2.2.14) and (2.2.15), at the points near the free boundary (e.g., point 1, Figure 5.4); and the mirror-image formula, (2.3.4), at those boundary gridpoints for which the normal derivative is specified. Observe that all the above formulae have a truncation error $O(h^4)$. The two resulting systems of equations are then solved for ϕ and ψ at the gridpoints of R by the point-Gauss-Seidel method.

In determining whether the appropriate, free boundary is used, the Cauchy-Riemann equations need be verified along only a curve in the interior of R. This follows

from the principle of analytic continuations (see Knopp [13], for example), in that, a function $w = \phi + i\psi$ which is analytic on a curve in the interior of R and for which $\nabla^2\phi = \nabla^2\psi = 0$ in R is analytic in all of R.

An alteration in the free boundary has the greatest effect on the Cauchy-Riemann equations for points near the free-boundary so that, the best choice of a curve on which the Cauchy-Riemann equations are to be established would be the free-boundary itself, assuming that $\nabla^2\phi = \nabla^2\psi = 0$ there. However, the point B is a stagnation point, and consequently, some difficulty is encountered in verifying the Cauchy-Riemann equations near B. To moderate the influence of the stagnation point and to facilitate numerical differentiation of ϕ and ψ, a curve consisting of broken-line segments above the free boundary, such as PQ in Figure 5.4, where the distance of PQ from the free boundary is at least h, is chosen. Standard differentiation formulae for equally-spaced points (see Appendix I) can then be applied at the gridpoints of the curve PQ.

It is assumed by intuition that the free boundary AB is a parabola of the form

$$x = (x_C - x_A)(y/y_B)^2 + x_A \,, \tag{5.2.1}$$

where $(x_A, 0)$ and (x_C, y_B) are the coordinates of the points A and B, respectively. Starting with initial guesses for x_A and y_B, these values are repeatedly modified so that for each alteration, the Cauchy-Riemann equations along PQ are better satisfied. A technique of improving the position of the free boundary is suggested in Table 5.3.

First Derivative	Points near P		Points near Q	
	Sign	Effect	Sign	Effect
$\partial\phi/\partial x$	+	↓	+	↓
$\partial\psi/\partial y$	+	↑	+	↑
$\partial\phi/\partial y$	-	↓	+	↕
$\partial\psi/\partial x$	+	↑	-	↓

Table 5.3

The Effects of Region Compression on the First Derivatives

[illegible]

[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

[illegible]

The table shows how an increase in the value of y_B ($y_B\uparrow$), or a decrease in the value of x_A ($x_A\downarrow$) (i.e., compression of the region R) influences the values of the first partial derivatives $\frac{\partial\phi}{\partial x}$, $\frac{\partial\psi}{\partial y}$, $\frac{\partial\phi}{\partial y}$, and $\frac{\partial\psi}{\partial x}$ at the points near P and Q . The symbol '$\updownarrow$' is used to indicate a negligible change of the derivative. When implementing this table to improve the position of the free boundary, it should be noted that the sign of a partial derivative changes at most one time along PQ ; and that, an alteration of x_A effects the partial derivatives most significantly at the point R . A similar observation holds true when adjusting the point B .

5.2.2 Data and Numerical Results A program written for the Fortran IV compiler to solve the canal problem in the physical plane (Problem 5.1) by the point-Gauss-Seidel method using finite-difference approximations of $O(h^4)$ can be found in Appendix IV. An initial estimate to ϕ and ψ in the region R is that ϕ varies linearly; and that, ψ varies parabolically with respect to x , and cubically with respect to y . More precisely, if y_J denotes the y-coordinate of the free boundary AB at $x_J = x_C + h(J-1)$, then

$$\psi(1,J) = \left(\frac{x_J - x_C}{x_D - x_C}\right)^2 ,$$

$$J=2,3,\ldots,K1 \quad \text{on} \quad CD ;$$

$$\phi(1,J) = -x_J/x_D ,$$

(5.2.2)

$$J=K1+1,K1+2,\ldots,K2-1 \quad \text{on} \quad DO ;$$

$$\psi(1,J) = \left(\frac{x_J}{x_A} - 1\right)^2 ,$$

$$J=K2,K2+1,\ldots,N \quad \text{on} \quad OA .$$

On lines of constant x (x_J, $J=1,2,\ldots,N$) ,

$$\phi(I,J) = \left(1 - \frac{y_I}{y_J}\right)\phi(1,J) + y_I ,$$

(5.2.3)

$$\psi(I,J) = (y_I - y_J)^2(2y_I + y_J)\psi(1,J)/y_J^2 ,$$

for $2 \le I \le M$ and $y_I > y_J$, given that $y_I = h(1-I)$.

The criterion for determining when to stop the point-Gauss-Seidel iteration is that the maximum relative deviation of u (u represents both ϕ and ψ) from the previous iteration is less than a preassigned tolerance e; that is, when

$$\max_{I,J} |(u^{k+1} - u^k)/u^{k+1}| < e . \tag{5.2.4}$$

For this investigation, e is chosen to be 0.005.

Test data are due to Charmonman [2] and include the three physical situations shown in Table 5.4. ϕ_C, x_C, are input parameters to the program which determine the dimensions of the problem to be solved; while x_A and y_B are the input parameters which determine the free boundary (5.2.1), and thus the region of fresh-water flow. The values of x_A and y_B are subsequently adjusted, by increments of at least 0.01, so as to minimize the maximum error in the Cauchy-Riemann equations; that is, minimize

$$\max_{x,y \text{ on } PQ} \left\{ \left|\frac{\partial\phi}{\partial x} - \frac{\partial\psi}{\partial y}\right|, \left|\frac{\partial\phi}{\partial y} + \frac{\partial\psi}{\partial x}\right| \right\} . \tag{5.2.5}$$

Case	1	2	3
ϕ_C	-4.0	-4.0	-2.0
x_C	-4.214	-5.549	-1.424
x_D	-3.893	-4.843	-1.848
x_A	0.50	0.50	0.48
y_B	-2.6	-3.0	-1.4
$(x_C-x_A)/y_B^2$	-0.697	-0.672	-0.971
Max. Error in Cauchy-Riemann Eqns.	0.40	0.38	0.33

Table 5.4

Test Data in the Physical Plane

The partial derivatives are calculated by use of numerical differentiation formulae of $O(h^2)$ given in Appendix I . Higher order formulae are not used because of the large

error, due to the inaccuracy of the finite-difference approximation, already inherent in the solution of ϕ and ψ. Adjusting the values of x_A and y_B is a tedious process and usually requires many runs on the computer. Charmonman [2] stated that about 18 runs were required to obtain the solution for each physical case.

The large error in the Cauchy-Riemann equations poses a question as to the accuracy of the solutions obtained. Although the values obtained for x_A and y_B correspond very closely with those obtained by Charmonman [2], some doubt remains as to the correct equation of the free boundary. Approximation of the free boundary by an ellipse produces even worse results, except for points near P. Various combinations of ellipses and parabolas prove to be just as futile.

Charmonman's method of obtaining the solution in the complex-potential plane and mapping it into the physical is more elegant and much more accurate. His method is demonstrated in the next section, for the problem of canals with intermediate drains.

5.3 Solution in the Complex-Potential Plane

5.3.1 Solution by Use of the Five-Point Formula

Laplace's equation for Problem 5.2 can be approximated by the five-point formula for the interior gridpoints of the

region R^* in Figure 5.3. We then have that

$$y(i,j) = \frac{1}{4}\,[y(i,j+1) + y(i-1,j) + y(i,j-1) + y(i+1,j)]$$

$$(5.3.1) \qquad j=1,2,\ldots,n\ , \quad i=1,2,\ldots,m\ .$$

For gridpoints along the boundary with the normal derivative specified (see Table 5.2), the mirror-image formula (2.3.4) yields

$$y(m+1,j) = \frac{1}{4}\,[y(m+1,j+1) + 2y(m,j) + y(m+1,j-1)]$$

$$(5.3.2) \qquad j=1,2,\ldots,k_1\ , \quad j=k_2,k_2+1,\ldots,n\ ;$$

while for gridpoints along the remaining segments of boundary, we have

$$
(5.3.3) \qquad
\begin{aligned}
y(i,0) &= 0 \,, & & \\
y(i,n+1) &= 0 \,, & & i=0,1,\ldots,m+1 \,, \\
y(0,j) &= 0 \,, & & j=1,2,\ldots,n \,, \\
y(m+1,j) &= -h(n+1-j) \,, & & j=k_1+1,k_2+2,\ldots,k_2-1 \,.
\end{aligned}
$$

Let I be the identity matrix of order n ; D , the square matrix of order n ,

$$
(5.3.4) \qquad D = \begin{bmatrix}
-4 & 1 & & & & & \\
1 & -4 & 1 & & & & \\
 & 1 & -4 & 1 & & & \\
 & & & \cdot & & & \\
 & & & & \cdot & & \\
 & & & & & \cdot & \\
 & & & & 1 & -4 & 1 \\
 & & & & & 1 & -4
\end{bmatrix} ;
$$

J , the matrix of dimension $n \times k_{12}$ ($k_{12} = n-k_2+k_1+1$) ,

$$(5.3.5) \qquad J = \begin{array}{cc}
\left[\begin{array}{cccc|cccc}
1 &&&&&&& \\
& 1 &&&&&& \\
&& \ddots &&&&& \\
&&& 1 &&&& \\ \hline
&&&&&&& \\
&&&&&&& \\ \hline
&&&& 1 &&& \\
&&&&& 1 && \\
&&&&&& \ddots & \\
&&&&&&& 1
\end{array}\right] & \begin{array}{c} 1 \\ 2 \\ \vdots \\ k_1 \\ \vdots \\ \vdots \\ k_2 \\ k_2+1 \\ \vdots \\ n \end{array} \\
\begin{array}{cccccccc} 1 & 2 & \cdots & k_1 & & & \cdots & k_{12} \end{array} &
\end{array} ;$$

K , the matrix of dimension $k_{12} \times n$,

$$(5.3.6) \qquad K = 2J^T ;$$

T , the matrix of order $k_{12} \times k_{12}$,

(5.3.7)

$$T = \begin{array}{c} \left[\begin{array}{cccc:cccc} -4 & 1 & & & & & & \\ 1 & -4 & 1 & & & & & \\ & & \ddots & & & & & \\ & & 1 & -4 & & & & \\ \hdashline & & & & -4 & 1 & & \\ & & & & 1 & -4 & 1 & \\ & & & & & & \ddots & \\ & & & & & & 1 & -4 \end{array}\right] \begin{array}{l} 1 \\ 2 \\ \vdots \\ k_1 \\ \\ \\ \vdots \\ k_{12} \end{array} \\ \begin{array}{cccc} 1 \quad 2 \quad \ldots \quad k_1 & & & \ldots \quad k_{12} \end{array} \end{array} ;$$

B_i $(i=1,2,\ldots,m)$, a column vector of dimension n ,

(5.3.8)

$$B_i = 0 \ , \quad i=1,2,\ldots,m-1 \ ,$$

$$B_m = [b_1, b_2, \ldots, b_n]^T \ ,$$

where

$$b_j = \begin{cases} 0 \ , & 1 \le j \le k_1 \ , \quad k_2 \le j \le n \ , \\ h(n+1,j) \ , & k_1 < j < k_2 \ ; \end{cases}$$

and B_{m+1} , a vector of dimension k_{12}

(5.3.9) $$B_{m+1} = [b_1', b_2', \ldots, b_{k_{12}}']^T ,$$

where

$$b_j' = \begin{cases} 0 , & j \neq k_1, k_1+1 , \\ b_{k_1}+1 , & j = k_1 , \\ b_{k_2}-1 , & j = k_1+1 . \end{cases}$$

If A denotes the block, tridiagonal matrix of order $mn+k_{12}$,

(5.3.10) $$A = \begin{bmatrix} D & I & & & & & & \\ I & D & I & & & & & \\ & I & D & I & & & & \\ & & & \cdot & & & & \\ & & & & \cdot & & & \\ & & & & & \cdot & & \\ & & & & & I & D & I & \\ & & & & & & I & D & J \\ & & & & & & & K & T \end{bmatrix} ,$$

and B the block vector of dimension $mn+k_{12}$,

$$B = [B_1^T, B_2^T, \ldots, B_{m+1}^T]^T ; \tag{5.3.11}$$

then the system of linear algebraic equations (5.3.1) and (5.3.2) can be written in matrix notation as

$$AY = B ; \tag{5.3.12}$$

where Y is the block vector of dimension $mn+k_{12}$,

$$Y = [Y_1^T, Y_2^T, \ldots, Y_{m+1}^T]^T , \tag{5.3.13}$$

with

$$Y_i \quad = [y(i,1),y(1,2),\ldots,y(i,n)]^T ,$$

$$i=1,2,\ldots,m , \tag{5.3.14}$$

$$Y_{m+1} = [y(m+1,1),y(m+1,2),\ldots,y(m+1,k_{12})]^T .$$

Problem 5.2 is now reduced to solving the system of equations (5.3.12).

5.3.1.1 Solution of System by the Point-Gauss-Seidel Method The system (5.3.12) can be solved by the point-Gauss-Seidel method by recursively applying the finite-difference formula (5.3.1) and (5.3.2) to the gridpoints of the region R^* (Figure 5.3). A consistent ordering of points, which corresponds to the ordering of points used for Y in (5.3.12), is given by

$$
\begin{aligned}
&(i,j)\ , && j=1,2,\ldots,n\ , \\
& && i=1,2,\ldots,m\ , \\
&(m+1,j)\ , && j=1,2,\ldots,k_1\ , \\
&(m+1,j)\ , && j=k_1+1,k_1+2,\ldots,n\ .
\end{aligned}
\tag{5.3.15}
$$

An initial guess to the solution Y is that $Y = 0$; and, a criterion for determining when to stop the point-Gauss-Seidel iteration is that the maximum deviation of Y from the previous iteration is less than a preassigned

tolerance e , that is, when

$$(5.3.16) \qquad \max_{i,j} |y^{k+1}(i,j) - y^{k}(i,j)| < e .$$

$mn+k_{12}$ multiplications and $3(mn+k_{12})$ additions are required for each point-Gauss-Seidel iteration.

5.3.1.2 Solution of System by the Gaussian Elimination Method Decomposition of the matrix A (5.3.10) into the product of a lower-triangular and an upper-triangular matrix by the Gaussian elimination method can be performed in blocks, as for the system (3.2.8) of the Torsion Problem 3.1 (see equation (3.2.14) and Figure 3.3 and 3.4). The complete process of forward-elimination and back-substitution requires approximately $(m+1)n^3$ multiplications, and about the same number of additions. A matrix of dimension $(mn+k_{12}) \times n$ is required to store the results of the forward-elimination of A .

5.3.1.3 Solution of System by Schechter's Method Schechter [18] describes a direct method of solving a system of equations, for which the coefficient matrix is of block, tridiagonal form, by means of partitioned matrices. His method, as applied to the system (3.2.12), is now presented.

If we factorize the matrix A (5.3.10) into the form

(5.3.17) $$A = LU ,$$

where

(5.3.18) $$L = \begin{bmatrix} I & & & & & & \\ L_2 & I & & & & & \\ & L_3 & I & & & & \\ & & \cdot & & & & \\ & & & \cdot & & & \\ & & & & \cdot & & \\ & & & & L_m & I & \\ & & & & & L_{m+1} & I \end{bmatrix}$$

and

(5.3.19) $$U = \begin{bmatrix} U_1 & I & & & & & \\ & U_2 & I & & & & \\ & & \cdot & & & & \\ & & & \cdot & & & \\ & & & & \cdot & & \\ & & & & U_{m-1} & I & \\ & & & & & U_m & J \\ & & & & & & U_{m+1} \end{bmatrix} ,$$

we find that the submatrices U_i and L_i are determined recursively as

$$U_1 = D ,$$

(5.3.20) $$U_i = D - P_{i-1}^{-1} , \quad i=2,3,\ldots,m ,$$

$$U_{m+1} = T - KP_m^{-1}J ;$$

and

$$L_i = P_{i-1}^{-1} , \quad i=2,3,\ldots,m ,$$

(5.3.21)

$$L_{m+1} = KP_m^{-1} ,$$

The solution to

(5.3.22) $$LZ = B ,$$

with Z defined as the block vector

(5.3.23) $$Z = [Z_1^T, Z_2^T, \ldots, Z_{m+1}^T]^T ,$$

is

$$Z_i = 0 , \quad i=2,3,\ldots,m-1 ,$$

(5.3.24) $$Z_m = B_m ,$$

$$Z_{m+1} = B_{m+1} - L_{m+1}B_m ,$$

where Z_i is a vector of dimension n for $i=1,2,\ldots,m$ and of dimension k_{12} for $i=m+1$. The solution Y to (5.3.12) can then be obtained from

(5.3.25) $$UY = Z ;$$

and is given by

$$Y_{m+1} = U_{m+1}^{-1} Z_{m+1} ,$$

(5.3.26) $$Y_m = U_m^{-1}(Z_m - JY_{m+1}) ,$$

$$Y_{m-1} = -U_{m-1}^{-1} Y_m ;$$

and the remaining Y_i $(i=m-2,m-3,\ldots,1)$ are determined recursively from

(5.3.27) $$Y_i = -DY_{i+1} - Y_{i+2} .$$

The number of inversions necessary to determine U_m can be reduced to one by setting

(5.3.28) $$U_i = P_{i-1}^{-1} P_i .$$

From (5.3.20), we then have

(5.3.29) $$P_i = P_{i-1}D - P_{i-2} , \quad i=2,3,\ldots,m$$

with

$$P_1 = D ,$$

$$P_0 = I .$$

Summarizing, the solution Y to (5.3.12) can be obtained by the following procedure:

1. Determine P_{m-2} and P_{m-1} using the recursive formula (5.3.29),

2. Calculate $U_{m-1}^{-1} = P_{m-1}^{-1}P_{m-2}$, using the method of Gaussian elimination to find P_{m-1}^{-1} ,

3. Calculate $U_m^{-1}(D - U_{m-1}^{-1})^{-1}$,

4. Calculate $L_{m+1} = KU_m^{-1}$,

5. Calculate $U_{m+1} = T - L_{m+1}J$,

6. Use the method of Gaussian elimination to solve

$$U_{m+1}Y_{m+1} = B_{m+1} - L_{m+1}B_m$$

for Y_{m+1} ,

7. Calculate $Y_m = U_m^{-1}(b_m - JY_{m+1})$,

8. Calculate $Y_{m-1} = -U_{m-1}^{-1} Y_m$,

9. Determine Y_i $(i=m-2,m-3,\ldots,1)$ using the recursive formula (5.3.27).

5.3.1.4 Numerical Results Computer programs to solve Problem 5.2, using the five-point formula to approximate Laplace's equation and solving the resulting system of equations by the three methods discussed in subsections (5.3.1.1), (5.3.1.2) and (5.3.1.3), can be found in Appendix V.

An example of Problem 5.2 is taken from Charmonman [3] in which $\phi_C = -4$. The values of ϕ_B and ϕ_A , which best solve the example (i.e., when (5.1.16) and (5.1.17) are best satisfied) is found to be -2.6 and -1.2), respectively. A flow net illustrating the solution to the problem is drawn in Section 5.3.3.

For the above example, the region R* (Figure 5.3) is superimposed by a square grid-system for mesh sizes of $\frac{1}{4}, \frac{1}{5}, \frac{1}{10}$. A summary of the numerical results, obtained for the y-coordinates at the gridpoints of each of the grid-systems, using the methods of (5.3.1.1), (5.3.1.2) and (5.3.1.3) is presented in Table 5.5, where the table entries are to be intrepreted as for Table 3.1.

Because two singularities, at the points A' and B , are present on the boundary of the region R* , the

Mesh Size	Number of Unknowns $mn+k_{12}$	Max. Truncation Error in Five-Point Formula	Comparison of First Two Methods				Schechter's Method	
			Max. Error in Solution of System	Time in Secs by Gauss. Elim. t_{G5}	Time in Secs by Point-Gauss-Seidel t_{S5}	$\frac{t_{S5}}{t_{G5}}$	Max. Error in Solution of System	Time in Secs
$\frac{1}{4}$	32	0.17	1.3×10^{-6}	0.13	0.15	1.15	6.5×10^{-3}	.23
$\frac{1}{5}$	87	0.11	1.6×10^{-5}	0.34	0.27	0.79	6.7×10^{-3}	.54
$\frac{1}{10}$	375	0.05	2.1×10^{-4}	7.05	2.82	0.40	7.2×10^{-2}	9.62

Table 5.5

Solution of the Canal Problem by the

Five-Point Formula

truncation error in approximating Laplace's equation by the five-point formula is very large at these points. The results obtained can, however, be improved be decreasing the interval size at these points; although no such effort is made in this investigation, since we are mainly interested in comparing the efficiency of direct and iterative methods for solving the systems of equations.

For approximations with mesh sizes $\frac{1}{4}$ and $\frac{1}{5}$, the time required to obtain solutions of the resulting systems of equations by the Gaussian elimination method are approximately equal to the times required to attain solutions with the same accuracy by the point-Gauss-Seidel method. For the smaller mesh, however, the point-Gauss-Seidel method is much faster. Since a better estimate to the solution than zero is probably possible, and because the accuracy of the solution obtained by the point-Gauss-Seidel method can be controlled, the point-Gauss-Seidel method is undoubtedly superior with respect to speed and accuracy to the Gaussian elimination method.

Using Schechter method, the results obtained are relatively much worse. Not only are the solutions to the systems less accurate, but the times required to obtain them much greater. The reason for the inaccuracy of the results, as explained by Schechter [18], is that rounding errors increase exponentially when using the recursive formulae (5.3.27) and (5.3.29).

5.3.2 Solution by Use of the Nine-Point Formula

Laplace's equation in Problem 5.2, for gridpoints in the interior of the region R^*, can be approximated by the nine-point formula, so that

$$y(i,j) = \frac{1}{20}\{4[y(i,j+1) + y(i+1,j) + y(i,j-1) + y(i-1,j)]$$

$$+ y(i+1,j+1) + y(i+1,j-1) + y(i-1,j-1) + y(i-1,j+1)\}\ ,$$

$$(5.3.30) \qquad j=1,2,\ldots,n\ ,$$

$$i=1,2,\ldots,m\ .$$

For gridpoints along the boundary with the normal derivative specified, we have

$$y(m+1,j) = \frac{1}{10}\{2[y(m+1,j+1) + 2y(m,j) + y(m+1,j-1)]$$

$$(5.3.31) \qquad + y(m,j+1) + y(m,j-1)\}\ ,$$

$$j=1,2,\ldots,k_1,k_2,\ldots,n\ ;$$

and for the remaining segments of the boundary, (5.3.3) holds true.

The system (5.3.30) and (5.3.31) can be written in the matrix notation (5.3.12) for the five-point formula; where now, however,

$$(5.3.32) \qquad I = \begin{bmatrix} 4 & 1 & & & & & \\ 1 & 4 & 1 & & & & \\ & 1 & 4 & 1 & & & \\ & & & \cdot & & & \\ & & & & \cdot & & \\ & & & & & \cdot & \\ & & & & 1 & 4 & 1 \\ & & & & & 1 & 4 \end{bmatrix};$$

$$(5.3.33) \quad D = \begin{bmatrix} -20 & 4 & & & & & \\ 4 & -20 & 4 & & & & \\ & 4 & -20 & 4 & & & \\ & & & \cdot & & & \\ & & & & \cdot & & \\ & & & & & \cdot & \\ & & & & 4 & -20 & 4 \\ & & & & & 4 & -20 \end{bmatrix};$$

(5.3.34)

$$J = \begin{array}{c} \left[\begin{array}{ccccc|ccccc} 4 & 1 & & & & & & & & \\ 1 & 4 & 1 & & & & & & & \\ & & \ddots & & & & & & & \\ & & 1 & 4 & 1 & & & & & \\ & & & 1 & 4 & & & & & \\ \hline & & & & 1 & & & & & \\ & & & & & 1 & & & & \\ \hline & & & & & 4 & 1 & & & \\ & & & & & 1 & 4 & 1 & & \\ & & & & & & & \ddots & & \\ & & & & & & & 1 & 4 & 1 \\ & & & & & & & & 1 & 4 \end{array}\right] \\ 1 \quad 2 \quad \cdots \quad k_1 \quad \cdots \quad k_{12} \end{array} \begin{array}{l} 1 \\ 2 \\ \vdots \\ \vdots \\ k_1 \\ \vdots \\ \vdots \\ k_2 \\ k_2+1 \\ \vdots \\ \vdots \\ n \\ \\ \end{array} ;$$

(5.3.35)

$$T = \begin{array}{c} \left[\begin{array}{cccc|cccc} -10 & 2 & & & & & & \\ 2 & -10 & 2 & & & & & \\ & & \ddots & & & & & \\ & & 2 & -10 & & & & \\ \hline & & & & -10 & 2 & & \\ & & & & 2 & -10 & 2 & \\ & & & & & & \ddots & \\ & & & & & & 2 & -10 \end{array}\right] \\ 1 \quad 2 \quad \cdots \quad k_1 \quad \cdots \quad k_{12} \end{array} \begin{array}{l} 1 \\ 2 \\ \vdots \\ k_1 \\ \vdots \\ \vdots \\ \vdots \\ k_{12} \\ \\ \end{array} ;$$

and

$$b_{k_1} \qquad h(n-k_1) \ ,$$

$$b_{k_1+1} = h[5(n-k_1)-1] \ ,$$

$$b_i \quad = 6h[n+1-i] \ , \quad i=k_1+2,k_1+3,\ldots,k_2-2 \ ,$$

$$b_{k_2-1} = h[5(n-k_2)+11] \ , \tag{5.3.36}$$

$$b_{k_2} \quad = h(n-k_2+2) \ ,$$

$$b'_{k_1} \quad = 2b_{k_1} \ ,$$

$$b'_{k_1+1} = 2b_{k_2} \ .$$

5.3.2.1 <u>Solution of System by the Point-Gauss-Seidel Method</u> After making an estimate to the solution Y to the system, (5.3.12), for the nine-point formula to be

zero, the estimate is improved recursively by applying formulae (5.3.30) and (5.3.31) to the gridpoints (5.3.15) of the region R^* . The point-Gauss-Seidel iteration is terminated when the condition (5.3.16) is satisfied. $2(mn+k_{12})$ multiplications, and $5mn + 5k_{12}$ additions, are required for each iteration.

5.3.2.2 Solution of System by the Gaussian Elimination Method As for the Torsion Problem 3.1, decomposition of the matrix A , (5.3.10) for the nine-point formula, can be performed in blocks (see equation (3.3.9) and Figures 3.5 and 3.6). Approximately $(m+1)n(n+1)^2$ multiplications, and the same number of additions, are required for the complete Gaussian elimination process. A matrix of dimension $(mn+k_{12}) \times (n+1)$ is necessary for storing the results of the forward-elimination of A .

5.3.2.3 Numerical Results Fortran IV programs to solve the system (5.3.12) for the nine-point formula, using the point-Gauss-Seidel method and the method of Gaussian elimination, can be found in Appendix V. A summary of the numerical results for mesh sizes of $\frac{1}{4}$, $\frac{1}{5}$ and $\frac{1}{10}$ is presented in Table 5.6, where the table entries are to be interpreted as for Table 3.2.

The point-Gauss-Seidel method is seen to be much faster

Mesh Size	Number of Unknowns $mn+k_{12}$	Max. Trun. Error in Nine-Point Formula	Max. Error in Solution of System	Time in Secs by Gauss. Elim. t_{G9}	Time in Secs by Point-Gauss-Seidel t_{S9}	$\frac{t_{S9}}{t_{G9}}$
$\frac{1}{4}$	32	0.15	1.7×10^{-5}	0.17	0.14	0.82
$\frac{1}{5}$	87	0.10	4.2×10^{-5}	0.43	0.29	0.67
$\frac{1}{10}$	375	0.03	3.4×10^{-4}	7.96	3.15	0.39

Table 5.6

Solution of the Canal Problem by the

Nine-Point Formula

than the Gaussian elimination method when solving the systems, for mesh sizes of $\frac{1}{5}$ and $\frac{1}{4}$, with the same accuracy. By choosing a better estimate to the solution than zero, or by allowing the accuracy of the solution of the system to correspond more closely with the accuracy of the finite-difference approximation, the point-Gauss-Seidel method becomes the more efficient method in all cases.

5.3.3 Summary of Results In Sections 3.3.1 and 3.3.2, we have discussed various methods of determining the y-coordinates of the gridpoints in the complex-potential plane (Figure 5.2), assuming that the values of $\phi_{A'}$ and ϕ_B for any particular ϕ_C are known. It remains to establish the accuracy of the choice of $\phi_{A'}$ and ϕ_B by calculating the x-coordinates of the gridpoints so that equations (5.1.16) and (5.1.17) are satisfied; and thereby, arriving at a complete solution to Problem 5.2.

The x-coordinates of the gridpoints are calculated by improvising the Cauchy-Riemann equations in the integral form (5.1.14) and (5.1.15). Starting from the point 0 at which $x = 0$, the x-coordinates are recursively calculated, being careful so as to minimize the influence of the singularities at the points A' and B. We, therefore, first determine x along the line OD of constant ψ, and proceed to calculate x along lines of constant ψ (line 34, Figure 5.3). Fortran IV programs using numerical differentiation

and integration formulae of $O(h^4)$ and of $O(h^6)$ (see Appendix I) are included in Appendix V.

For $\phi_C = -4$, we determine $\phi_{A'}$ and ϕ_B to be -1.2 and -2.6, respectively, corresponding to the physical situation with $x_C = -4.214$ and $x_B = -3.893$. A flow net, drawn to scale is shown in Figure 5.5.

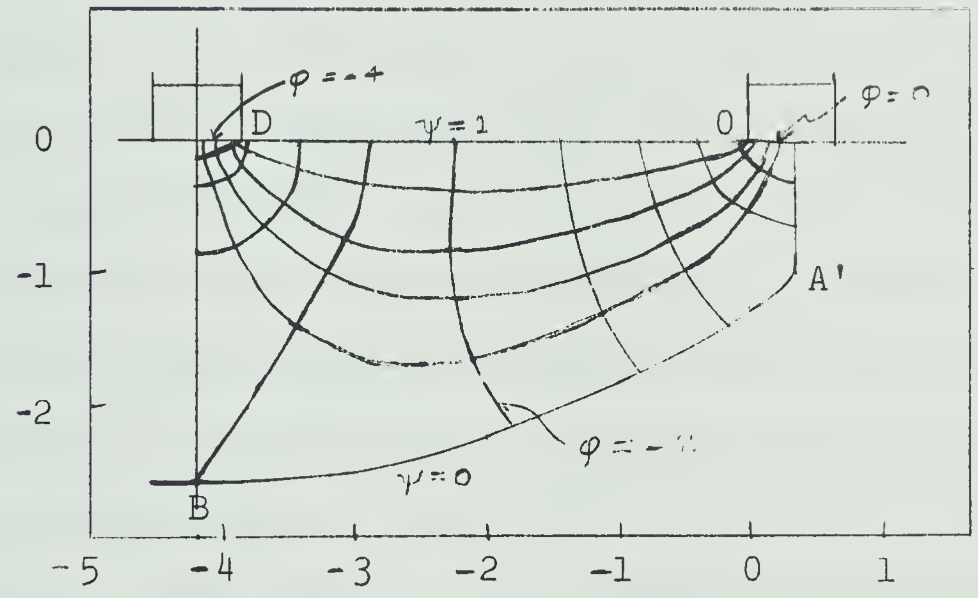

Figure 5.5 Flow Net for the Canal Problem

CHAPTER VI

CONCLUDING REMARKS AND SUGGESTIONS FOR FUTURE RESEARCH

We have considered three examples of Laplace's equation: namely, the Torsion Problem 3.1 with no singularities in the region under consideration; the Channel Problem 4.1 with one singularity on the boundary of the region; and the Canal Problem 5.2 with two singularities on the boundary of the region. The five-point formula and the nine-point formula are used to approximate Laplace's equation, with no attempt made to reduce the influence of the singularities on the accuracy of the finite-difference formulae.

A favored but somewhat inelegant method of coping with a singularity is effectively to ignore it, and to mitigate its effect (the effect of the unboundedness of the partial derivatives of some order, on the finite-difference formulae) by using a smaller mesh size in the neighbourhood of the singularity. A more elegant technique is to subtract out the singularity; that is, for a function u which is singular at some point(s) in a region, or on the boundary of the region, find a function w so that $u-w$ is analytic in the whole region (see, for example, Motz [15] and Woods [25]). Such a function w, however, cannot always be found. It might be interesting to attempt to reduce the

effect of a singularity by making a transformation of a form such as

$$v = e^{-(u+c)} ,$$

where c is a large, positive constant. Laplace's equation for u then becomes Poisson's equation

$$\nabla^2 v = e^{-(u+c)}\left[\left(\frac{\partial u}{\partial x}\right)^2 + \left(\frac{\partial u}{\partial y}\right)^2\right]$$

When solving the systems of equations resulting from the finite-difference approximations, we have shown that, for the latter two examples, the point-Gauss-Seidel method is undoubtedly the more efficient method, with respect to speed and accuracy, as compared to the Gaussian elimination method. For the first example, however, the rate of convergence (determined by the rate with which $(\max_i \lambda_i)^k \to 0$ as $k \to \infty$, where the λ_i are the eigenvalues of the coefficient matrix of the system (see Forsythe and Wasow [8])) of the point-Gauss-Seidel method appears to be much smaller, so that the Gaussian elimination method may be the more efficient method of solving the problem (for some

particular mesh sizes, at least). In fact, for case (*) in Table 3.3 with a large number of gridpoints, a moderate increase in the accuracy requirements for the solution of the system demanded a large increase in the number of point-Gauss-Seidel iterations (3.36 times as many). A possibility for future research thereby suggests itself, in that, for large systems of equations where the largest eigenvalue of the coefficient matrix is close to unity, it might be faster to obtain the solution to a specified accuracy (governed by the accuracy of the finite-difference approximation) by using the method of Gaussian elimination with iterative improvement (see Ralston [16]).

As already mentioned in Chapter I, iterative methods faster than the point-Gauss-Seidel method have been developed; but, a necessary condition for the convergence of all of these methods is that the point-Gauss-Seidel method be convergent. Thus, for systems where the convergence of the point-Gauss-Seidel method is slow (i.e., when the largest eigenvalue in magnitude is close to unity), the Gaussian elimination method might be faster than any iterative method; while for other systems, iterative methods may not converge at all. For Problem 3.1, however, the method of successive over-relaxation [8] with the proper choice of an over-relaxation factor (see Young [26] for the optimum over-relaxation factor of a square region

with Dirichlet boundary conditions) should be faster than the Gaussian elimination method.

Direct methods other than the method of Gaussian elimination have also been developed. Included are the methods of Cornock [5], Schechter [18] and Wilson [23], which are based on the use of partitioned matrices. These methods all require the calculation of the inverses of a number of submatrices, so that the solutions obtained by these methods will be inaccurate if these submatrices are ill-conditioned. Furthermore, as pointed out in Fox [10], round-off errors incurred during the calculations are propogated exponentially.

Since Schechter's method requires the least number of inversions (only three, see Section 5.3.3), an attempt to solve Problem 5.2 using his method was made. The results obtained are much worse, with respect to time and accuracy, than those obtained by the Gaussian elimination method. There is little hope that the methods due to Cornock and Wilson will produce better results.

BIBLIOGRAPHY

1. Bickley, W.G., "Finite Difference Formulae for the Square Lattice", Q.J. Mech. and Appl. Math., 1 (1948), p. 35.

2. Charmonman, S., "A Numerical Method of Solution of Free Surface Problems", Journal of Geophysical Research, 71 (1966), pp. 3861-3868.

3. Charmonman, S., "Coastal Parallel Canals with Intermediate Drains", Journal of the Hydraulics Division, ASCE, 93 (1967), pp. 5053-5067.

4. Collatz, L., The Numerical Treatment of Differential Equations, Springer-Verlag, Inc., New York, 1966.

5. Cornock, A.F., "The Numerical Solution of Poisson's and the Biharmonic Equation by Matrices", Proc. Camb. Phil. Soc., 50 (1954), pp. 524-535.

6. Davis, P.J., Interpolation and Approximation, Blaisdell Publishing Co., New York, 1963.

7. Duff, G.F.D. and D. Naylor, Differential Equations of Applied Mathematics, John Wiley and Sons, Inc., New York, 1966.

8. Forsythe, G.E. and W.R. Wasow, Finite-Difference Methods for Partial Differential Equations, John Wiley and Sons, Inc., New York, 1967.

9. Fox, L., "Solution by Relaxation Methods of Plane Potential Problems with Mixed Boundary Conditions", Quart. Appl. Math., 2 (1944), pp. 251-257.

10. Fox, L., Numerical Solution of Ordinary and Partial Differential Equations, Addison-Wesley Publishing, Co., Inc., Reading, Mass., 1962.

11. Greenspan, D., Introduction to Partial Differential Equations, McGraw-Hill Book Co., Inc., New York, 1961.

12. Hildebrand, F.B., Advanced Calculus for Applications, Prentice-Hall, Inc., Englewood Cliffs, N.J., 1963.

13. Knopp, K., Theory of Functions, Pt. 1, Dover Publications, Inc., New York, 1945.

14. Milne, W.E., Numerical Calculus, Princeton University Press, Princeton, N.J., 1949.

15. Motz, H., "The Treatment of Singularities of Partial Differential Equations by Relaxation Methods", Quart, Appl. Math., 4 (1946), pp. 371-377.

16. Ralston, A., A First Course in Numerical Analysis, McGraw-Hill Book Co., Inc., New York, 1965.

17. Sagan, H., Boundary and Eigenvalue Problems in Mathematical Physics, John Wiley and Sons, Inc., New York, 1963.

18. Schechter, S., "Quasi-tridiagonal Matrices and Type-insensitive Difference Equations", Quart, Appl. Math., 18 (1960), pp. 285-295.

19. Sobelev, S.L., Partial Differential Equations of Mathematical Physics, Addison-Wesley Publishing Co., Inc., Reading, Mass., 1964.

20. Thom, A. and C.J. Apelt, Field Computations in Engineering and Physics, J.W. Arrowsmith Ltd., Bristol, Eng., 1961.

21. Varga, R.S., Matrix Iterative Analysis, Prentice-Hall, Inc., Englewood Cliffs, N.J., 1962.

22. Viswanathan, R.V., "Solution of Poisson's Equation by Relaxation Methods - Normal Gradient Specified on Curved Boundaries", Math. Tab., Wash., 11 (1957), pp. 67-78.

23. Wilson, L.B., "Solution of Certain Large Sets of Equations on Pegasus Using Matrix Methods", Computer J., 2 (1959), pp. 130-133.

24. Winograd, S., "A New Algorithm for Inner Product", IBM Watson Research Center, Research Report RC-1943, Yorktown Heights, N.Y., November 21, 1967.

25. Woods, L.C., "The Relaxation Treatment of Singular Points in Poisson's Equation", Quart, J. Mech., 6 (1953), pp. 163-185.

26. Young, D., "Iterative Methods for Solving Partial Differential Equations of Elliptic Type", Transactions, AMS, 76 (1954), pp. 93-111.

APPENDIX I

NUMERICAL DIFFERENTIATION AND INTEGRATION

Numerical differentiation or integration formulae can be derived by interpolating the function $y = f(x)$, whose derivative or integral we seek, by a polynomial of sufficient degree; and then, differentiating or integrating the latter. This method of deriving the formulae is often called the method of undetermined coefficients (see Milne [14]).

The truncation error incurred in all cases can be obtained by making use of the following theorem due to Peano (see Davis [6]):

<u>Theorem I.1</u> *Let* $L(p) = 0$ *for all* $p \varepsilon P_n$, *where* L *denotes a linear function,* p *an arbitrary polynomial, and* P_n *the set of all polynomials of degree* n *or less. Then for all* $f \varepsilon C^{n+1}[a,b]$,

$$L(f) = \int_a^b f^{(n+1)}(t)\, K(t)dt ,$$

where

$$K(t) = \frac{1}{n!} L_x[(x-t)_+^n]$$

and

$$(x-t)_+^n = \begin{cases} (x-t)^n & x \geq t\ , \\ 0 & x < t\ . \end{cases}$$

If in addition the kernel $K(t)$ *does not change sign on* $[a,b]$ *, then*

$$L(f) = \frac{f^{(n+1)}(\nu)}{(n+1)!} L(x^{n+1})\ , \quad a \leq \nu \leq b\ .$$

Note that the differentiation and integration operations are linear functionals when $f \varepsilon C^{n+1}[a,b]$.

Numerical differentiation and integration formulae for equally-spaced points are now listed without proof. We assume that

$$y_i = f(x_i)\ ,$$

and that

$$x_i = x_0 + ih\ , \quad i=1,2,\ldots,6\ ,$$

where h is the interval size.

I.1 Differentiation Formulae

I.1.1 Formulae of $O(h^2)$

$$y_0' = \frac{1}{2h}(-3y_0 + 4y_1 - y_2) + \frac{h^2}{3} f^{(3)}(\nu_0)$$

$$y_1' = \frac{1}{2h}(-y_0 + y_2) - \frac{h^2}{6} f^{(3)}(\nu_1)$$

$$y_2' = \frac{1}{2h}(y_0 - 4y_1 + 3y_2) + \frac{h^2}{3} f^{(3)}(\nu_2)$$

$$x_0 \le \nu_i \le x_2$$

I.1.2 Formulae for $O(h^4)$

$$y_0' = \frac{1}{12h}(-25y_0 + 48y_1 - 36y_2 + 16y_3 - 3y_4) + \frac{h^4}{5} f^{(5)}(\nu_0)$$

$$y_1' = \frac{1}{12h}(-3y_0 - 10y_1 + 18y_2 - 6y_3 + y_4) - \frac{h^4}{20} f^{(5)}(\nu_1)$$

$$y_2' = \frac{1}{12h}\left(y_0 - 8y_1 + 8y_3 - y_4\right) + \frac{h^4}{30} f^{(5)}(\nu_2)$$

$$y_3' = \frac{1}{12h}\left(- y_0 + 6y_1 - 18y_2 + 10y_3 - 3y_4\right) - \frac{h^4}{20} f^{(5)}(\nu_3)$$

$$y_4' = \frac{1}{12h}\left(3y_0 - 16y_1 + 36y_2 - 48y_3 + 25y_4\right) + \frac{h^4}{5} f^{(5)}(\nu_4)$$

$$x_0 \le \nu_i \le x_4$$

I.1.3 Formulae of $O(h^6)$

$$y_0' = \frac{1}{60h}\left(-147y_0 + 360y_1 - 450y_2 + 400y_3 - 225y_4 + 75y_5 - 10y_6\right) + \frac{h^6}{7} f^{(7)}(\nu_0)$$

$$y_1' = \frac{1}{60h}\left(- 10y_0 - 77y_1 + 150y_2 - 100y_3 + 50y_4 - 15y_5 + 2y_6\right) - \frac{h^6}{42} f^{(7)}(\nu_1)$$

$$y_2' = \frac{1}{60h}(\ 2y_0 - 24y_1 - 35y_2 + 80y_3 - 30y_4$$

$$+ 8y_5 - y_6) + \frac{h^6}{105} f^{(7)}(\nu_2)$$

$$y_3' = \frac{1}{60h}(- y_0 + 9y_1 - 45y_2 + 45y_4$$

$$- 9y_5 + y_6) - \frac{h^6}{190} f^{(7)}(\nu_3)$$

$$y_4' = \frac{1}{60h}(\ y_0 - 8y_1 + 30y_2 - 80y_3 + 35y_4$$

$$+ 25y_5 - 2y_6) + \frac{h^6}{105} f^{(7)}(\nu_4)$$

$$y_5' = \frac{1}{60h}(- 2y_0 + 15y_1 - 50y_2 + 100y_3 - 150y_4$$

$$+ 77y_5 + 10y_6) - \frac{h^6}{42} f^{(7)}(\nu_5)$$

$$y_6' = \frac{1}{60h}(\ 10y_0 - 72y_1 + 225y_2 - 400y_3 + 450y_4$$

$$- 360y_5 + 147y_6) + \frac{h^6}{7} f^{(7)}(\nu_6)$$

I.2 Integration Formulae

I.2.1 Formulae of $O(h^4)$

$$\int_{x_0}^{x_2} f(x)dx = \frac{h}{3}\ (y_0 + 4y_1 + y_2) - \frac{h^5}{90} f^{(4)}(\nu_0)$$

$$\int_{x_1}^{x_2} f(x)dx = \frac{h}{12}\ (-y_0 + 8y_1 + 5y_2) - \frac{h^4}{24} f^{(3)}(\nu_1)$$

$$x_0 \le \nu_i \le x_2$$

I.2.2 Formulae of $O(h^6)$

$$\int_{x_0}^{x_2} f(x)dx = \frac{4h}{90}\ (7y_0 + 32y_1 + 12y_2 + 32y_3 + 7y_4)$$

$$- \frac{8h^7}{945} f^{(6)}(\nu_0)$$

$$\int_{x_1}^{x_4} f(x)dx = \frac{h}{240}\ (- 9y_0 + 126y_1 + 216y_2 + 306y_3 + 81y_4)$$

$$- \frac{9}{4} h^6 f^{(5)}(\nu_1)$$

$$\int_{x_2}^{x_4} f(x)dx = \frac{h}{90} (- \ y_0 + \ 4y_1 + \ 24y_2 + 124y_3 + \ 29y_4)$$

$$-\frac{4}{3} h^6 \ f^{(5)}(\nu_2)$$

$$\int_{x_3}^{x_4} f(x)dx = \frac{h}{720} (-19y_0 + 106y_1 - 264y_2 + 646y_3 + 251y_4)$$

$$-\frac{9h^6}{4} f^{(5)}(\nu_3)$$

$$x_0 \le \nu_i \le x_4$$

APPENDIX II

PROGRAM LISTINGS FOR THE TORSION PROBLEM

This section contains program listings for the Fortran IV compiler for the IBM 360/67 computing system to solve the Torsion Problem 3.1. Included are programs to solve, by the point-Gauss-Seidel and the Gaussian elimination methods, the systems resulting when the five-point formula and the nine-point formula are used to approximate Laplace's equations (see Chapter III).

```
C       LAPLACE'S EQUATION FOR THE TORSION PROBLEM
C       IS APPROXIMATED BY THE FIVE-POINT FORMULA.
C       THE RESULTING SYSTEM IS SOLVED BY
C       THE GAUSS-SEIDEL METHOD.
C
        DIMENSION  U(41,41)
        READ(5,1) N,H,ERR
      1 FORMAT(I10,2F10.0)
C
C       AN INITIAL GUESS IS ZERO.
C       BOUNDARY CONDITIONS ARE X**2 + Y**2.
C
        NN=N+1
        HH=H*H
        DO 3 I=1,N
        DO 2 J=1,N
        U(I,J)=0.
      2 CONTINUE
        TEMP=1.+HH*(I-1)**2
        U(I,NN)=TEMP
        U(NN,I)=TEMP
      3 CONTINUE
C
C       APPLY GAUSS-SEIDEL ITERATION.
C
        L=0
        DO 7 K=1,1000
        L=L+1
        ERROR=0.
        TEMP=0.5*(U(1,2)+U(2,1))
        ERROR=AMAX1(ERROR,ABS(U(1,1)-TEMP))
        U(1,1)=TEMP
        DO 4 J=2,N
        TEMP=0.25*(U(1,J-1)+U(1,J+1)+U(2,J)+U(2,J))
        ERROR=AMAX1(ERROR,ABS(U(1,J)-TEMP))
        U(1,J)=TEMP
      4 CONTINUE
        DO 6 I=2,N
        TEMP=0.25*(U(I-1,1)+U(I+1,1)+U(I,2)+U(I,2))
        ERROR=AMAX1(ERROR,ABS(U(I,1)-TEMP))
        U(I,1)=TEMP
        DO 5 J=2,N
        TEMP=0.25*(U(I,J+1)+U(I+1,J)+U(I,J-1)+U(I-1,J))
        ERROR=AMAX1(ERROR, ABS(U(I,J)-TEMP))
        U(I,J)=TEMP
      5 CONTINUE
      6 CONTINUE
        IF(ERROR.LE.ERR) GO TO 9
      7 CONTINUE
        WRITE(6,8)
```

```
  8 FORMAT (' NO CONVERGENCE')
  9 WRITE(6,10) L,ERROR
 10 FORMAT(I10,E15.3)
    DO 12 I=1,N
    WRITE(6,11) (U(I,J),J=1,N)
 11 FORMAT(1X,10F12.7)
 12 CONTINUE
    STOP
    END
```

```
C       GAUSSIAN ELIMINATION IS IMPLEMENTED TO FIND
C       THE SOLUTION TO LAPLACE'S EQUATION FOR
C       THE TORSION PROBLEM. LAPLACE'S EQUATION IS
C       APPROXIMATED BY THE FIVE-POINT FORMULA.
C
        DIMENSION A(400,20),B(400)
        COMMON A,B
        READ(5,1) H,N
      1 FORMAT(F10.0,I10)
        HH=H*H
        NN=N-1
        I1=N*N
        I2=I1-N
        I3=I2+1
C
C       GENERATION OF THE RIGHT-HAND SIDE B.
C
        K=0
        DO 3 I=1,NN
        DO 2 J=1,NN
        K=K+1
        B(K)=0.
      2 CONTINUE
        K=K+1
        B(K)=-(1.+HH*(I-1)**2)
        B(I+I2)=B(K)
      3 CONTINUE
        B(I1)=-2.*((H-2)*H+2.)
C
C       GENERATION OF THE FIRST BLOCK.
C
        DO 6 I=1,N
        DO 5 J=1,N
        A(I,J)=0.
      5 CONTINUE
        A(I,I)=-4.
        IF(I.NE.1) A(I,I-1)=1.
        IF(I.NE.N) A(I,I+1)=1.
      6 CONTINUE
        A(1,2)=2.
C
C       DECOMPOSITION OF THE FIRST N-1 BLOCKS.
C
        DO 7 I=1,NN
        CALL TOR5(I,N)
      7 CONTINUE
C
C       DECOMPOSITION OF THE N'TH BLOCK
C
        I4=I1-1
```

```
      NE=N
      DO 10 I=I3,I4
      I5=I+1
      TP=-1./A(I,1)
      DO 9 J=I5,I1
      TEMP=TP*A(J,1)
      DO 8 K=2,NE
      A(J,K-1)=A(J,K)+TEMP*A(I,K)
    8 CONTINUE
      B(J)=B(J)+TEMP*B(I)
    9 CONTINUE
      NE=NE-1
   10 CONTINUE
C
C     BACK-SUBSTITUTION ON THE N'TH BLOCK.
C
      A(I1,1)=B(I1)/A(I1,1)
      I5=I1+1
      DO 12 L=2,N
      I=I5-L
      TEMP=B(I)
      DO 11 K=2,L
      TEMP=TEMP-A(I,K)*A(I+K-1,1)
   11 CONTINUE
      A(I,1)=TEMP/A(I,1)
   12 CONTINUE
C
C     BACK-SUBSTITUTION ON REMAINING BLOCKS.
C
      DO 14 L=1,I2
      I=I3-L
      K=I+N
      TEMP=B(I)-A(K,1)
      IF(I.LE.N) TEMP=TEMP-A(K,1)
      K=I-1
      DO 13 J=2,N
      TEMP=TEMP-A(I,J)*A(K+J,1)
   13 CONTINUE
      A(I,1)=TEMP/A(I,1)
   14 CONTINUE
      WRITE(6,15)(A(I,1),I=1,I1)
   15 FORMAT(10F12.7)
      STOP
      END
```

```
      SUBROUTINE TORS(L,N)
C
C     DECOMPOSITION OF THE K'TH BLOCK FOR
C     THE TORSION PROBLEM. LAPLACE'S EQUATION
C     IS APPROXIMATED BY THE FIVE-POINT FORMULA.
C
      DIMENSION A(400,20),B(400)
      COMMON A,B
      NN=N-1
      IE=L*N
      IN=IE-NN
      IN1=IN+1
      DO 4 I=IN,IE
      I1=I+1
      I2=I+NN
      TP=-1./A(I,1)
      DO 2 J=I1,I2
      TEMP=TP*A(J,1)
      DO 1 K=2,N
      A(J,K-1)=A(J,K)+TEMP*A(I,K)
    1 CONTINUE
      B(J)=B(J)+TEMP*B(I)
      IF(L.EQ.1) TEMP=TEMP+TEMP
      A(J,N)=TEMP
    2 CONTINUE
      IF(I.NE.IN) A(I2,N)=A(I2,N)+1
      IF(I.EQ.IN1) A(I2,N)=A(I2,N)+1
      I3=I2+1
      DO 3 K=2,N
      A(I3,K-1)=TP*A(I,K)
    3 CONTINUE
      IF(I.NE.IN) A(I3,NN)=A(I3,NN)+1.
      A(I3,N)=TP-4.
      IF(L.EQ.1) A(I3,N)=A(I3,N)+TP
      B(I3)=B(I3)+TP*B(I)
    4 CONTINUE
    5 RETURN
      END
```

```
C       LAPLACE'S EQUATION FOR THE TORSION PROBLEM
C       IS APPROXIMATED BY THE NINE-POINT FORMULA.
C       THE RESULTING SYSTEM IS SOLVED BY THE
C       GAUSS-SEIDEL METHOD
C
        DIMENSION U(41,41)
        READ(5,1) N,H,ERR
      1 FORMAT(I10,2F10.0)
        NN=N+1
        HH=H*H
C
C       AN INITIAL GUESS IS ZERO.
C       BOUNDARY CONDITIONS ARE X**2+Y**2.
C
        DO 3 I=1,N
        DO 2 J=1,N
        U(I,J)=0.
      2 CONTINUE
        TEMP=1.+HH*(I-1)**2
        U(I,NN)=TEMP
        U(NN,I)=TEMP
      3 CONTINUE
        U(NN,NN)=2.
C
C       APPLY GAUSS-SEIDEL ITERATION
C
        L=0
        DO 7 K=1,1000
        L=L+1
        ERROR=0.
        TEMP=0.2*(U(1,2)+U(1,2)+U(2,2)+U(2,1)+U(2,1))
        ERROR=AMAX1(ERROR,ABS(U(1,1)-TEMP))
        U(1,1)=TEMP
        DO 4 J=2,N
        TEMP=0.1*(2.*U(1,J+1)+U(2,J+1)+4.*U(2,J)
       1+ U(2,J-1)+2.*U(1,J-1))
        ERROR=AMAX1(ERROR,ABS(U(1,J)-TEMP))
        U(1,J)=TEMP
      4 CONTINUE
        DO 6 I=2,N
        TEMP=0.1*(2.*U(I-1,1)+U(I-1,2)+4.*U(I,2)
       1+U(I+1,2)+2.*U(I+1,1))
        ERROR=AMAX1(ERROR,ABS(U(I,1)-TEMP))
        U(I,1)=TEMP
        DO 5 J=2,N
        TEMP=0.05*(4.*(U(I,J+1)+U(I+1,J)+U(I,J-1)+U(I-1,J))
       1+U(I-1,J+1)+U(I+1,J+1)+U(I+1,J-1)+U(I-1,J-1))
        ERROR=AMAX1(ERROR,ABS(U(I,J)-TEMP))
        U(I,J)=TEMP
      5 CONTINUE
```

```
 6 CONTINUE
   IF(ERROR.LE.ERR) GO TO 9
 7 CONTINUE
   WRITE(6,8)
 8 FORMAT(' NO CONVERGENCE')
9  CONTINUE
 9 WRITE(6,10) L,ERROR
10 FORMAT(I10,E15.8)
   DO 12 I=1,N
   WRITE(6,11)(U(I,J),J=1,N)
11 FORMAT(1X,10F12.7)
12 CONTINUE
   STOP
   END
```

```
C       LAPLACE'S EQUATION FOR THE TORSION
C       PROBLEM IS APPROXIMATED BY THE NINE-POINT
C       FORMULA. THE RESULTING SYSTEM IN SOLVED
C       BY THE GAUSSIAN ELIMINATION METHOD.
C
        DIMENSION A(400,21),B(400)
        COMMON A,B
        READ(5,1) H,N
      1 FORMAT (F10.0,I10)
        N0=N-1
        I1=N*N
        N1=N+1
        I2=I1-N
        I3=I2+1
C
C       GENERATION OF THE RIGHT-HAND SIDE B.
C
        HH=H*H
        K=0
        DO 3 I=1,N0
        DO 2 J=1,N0
        K=K+1
        B(K)=0.
      2 CONTINUE
        K=K+1
        B(K)=-(6.+HH*((6.*I-12.)*I+8.))
        B(I+I2)=B(K)
      3 CONTINUE
        B(N)=0.5*B(N)
        B(I1)=(24.-16.*H)*H-24.
C
C       GENERATION OF THE FIRST BLOCK
C
        DO 5 I=1,N
        DO 4 J=1,N
        A(I,J)=0.
      4 CONTINUE
        A(I,I)=-10.
        IF(I.NE.1) A(I,I-1)=2.
        IF(I.NE.N) A(I,I+1)=2.
      5 CONTINUE
        A(1,2)=4.
C
C       DECOMPOSITION OF THE FIRST N-1 BLOCKS
C
        DO 6 I=1,N0
        CALL TOR9(I,N)
      6 CONTINUE
C
C       DECOMPOSITION OF THE N'TH BLOCK.
```

```
C
      I4=I1-1
      NE=N
      DO 9 I=I3,I4
      I5=I+1
      TP=-1./A(I,1)
      DO 8 J=I5,I1
      TEMP=TP*A(J,1)
      DO 7 K=2,NE
      A(J,K-1)=A(J,K)+TEMP*A(I,K)
    7 CONTINUE
      B(J)=B(J)+TEMP*B(I)
    8 CONTINUE
      NE=NE-1
    9 CONTINUE
C
C     BACK SUBSTITUTION ON THE N'TH BLOCK.
C
      A(I1,1)=B(I1)/A(I1,1)
      I5=I1+1
      DO 11 L=2,N
      I=I5-L
      TEMP=B(I)
      DO 10 K=2,L
      TEMP=TEMP-A(I,K)*A(I+K-1,1)
   10 CONTINUE
      A(I,1)=TEMP/A(I,1)
   11 CONTINUE
C
C     BACK-SUBSTITUTION ON REMAINING BLOCKS
C
      I=I3
      DO 14 L=1,NO
      DO 13 LL=1,N
      I=I-1
      TEMP=B(I)
      IF(LL.NE.1) TEMP=TEMP-A(I+N1,1)
      IF(LL.EQ.N) TEMP=TEMP-A(I+M1,1)
      DO 12 K=1,N
      TEMP=TEMP-A(I,K+1)*A(I+K,1)
   12 CONTINUE
      A(I,1)=TEMP/A(I,1)
   13 CONTINUE
   14 CONTINUE
      WRITE(6,15)(A(I,1),I=1,I1)
   15 FORMAT(1X,10F12.7)
      STOP
      END
```

```
      SUBROUTINE TOR9(L,N)
C
C     DECOMPOSITION OF THE L'TH BLOCK FOR THE
C     TORSION PROBLEM. LAPLACE'S EQUATION IS
C     APPROXIMATED BY THE NINE-POINT FORMULA.
C
      DIMENSION A(400,21),B(400)
      COMMON A,B
      NO=N-1
      N1=N+1
      I2=L*N
      I3=I2-1
      I=I2-NO
      I1=I+1
C
C     ELIMINATION OF THE FIRST COLUMN.
C
      A(I,N1)=4.
      TP=-1./A(I,1)
      DO 2 J=I1,I2
      TEMP=TP*A(J,1)
      DO 1 K=2,N
      A(J,K-1)=A(J,K)+TEMP*A(I,K)
    1 CONTINUE
      B(J)=B(J)+TEMP*B(I)
      TEMP=TEMP+TEMP
      A(J,N1)=TEMP
      A(J,N)=TEMP+TEMP
    2 CONTINUE
      A(I1,N)=A(I1,N)+1
      A(I1,N1)=A(I1,N1)+4.
      A(I+2,N1)=A(I+2,N1)+1.
      NI=I2+1
      NI1=NI+1
      TEMP=4.*TP
      DO 3 K=2,N
      A(NI,K-1)=TEMP*A(I,K)
      A(NI1,K-1)=TP*A(I,K)
    3 CONTINUE
      A(NI,1)=A(NI,1)+2.
      A(NI1,1)=A(NI1,1)+4.
      A(NI1,2)=A(NI1,2)+1.
      A(NI,N)=4.*TEMP-20.
      A(NI,N1)=2.*TEMP+8.
      A(NI1,N)=TEMP+4.
      A(NI1,N1)=2.*TP-20.
      B(NI)=TEMP*B(I)
      B(NI1)=TP*B(I)
C
C     ELIMINATION OF REMAINING COLUMNS
```

```
      DO 7 I=I1,I2
      I1=I+1
      NI=I+N
      NI1=NI+1
      TP=-1./A(I,1)
      DO 5 J=I1,NI
      TEMP=TP*A(J,1)
      DO 4 K=2,N1
      A(J,K-1)=A(J,K)+TEMP*A(I,K)
    4 CONTINUE
      IF(I.NE.I2) A(J,N1)=TEMP
      B(J)=B(J)+TEMP*B(I)
    5 CONTINUE
      IF(I.EQ.I2) GO TO 8
      A(I1,N1)=A(I1,N1)+4.
      IF(I.NE.I2) A(I+2,N1)=A(I+2,N1)+1.
      A(NI,N1)=A(NI,N1)+4.
      A(NI1,1)=TP*A(I,2)+4.
      DO 6 K=3,N1
      A(NI1,K-1)=TP*A(I,K)
    6 CONTINUE
      IF(I.NE.I3)A(NI1,2)=A(NI1,2)+1.
      A(NI1,N)=A(NI1,N)+4.
      A(NI1,N1)=TP-20.
      B(NI1)=B(NI1)+TP*B(I)
    7 CONTINUE
    8 RETURN
      END
```

APPENDIX III

PROGRAM LISTINGS FOR THE CHANNEL PROBLEM

Included in this section are Fortran IV program listings to solve, by the point-Gauss-Seidel and the Gaussian elimination methods, the systems resulting when Laplace's equation for the Channel Problem 4.1 is approximated by the five-point and the nine-point formulae. The procedure that is used in each case is described in Chapter IV.

```
C       LAPLACE'S EQUATION IN AN L-SHAPED CHANNEL IS
C       APPROXIMATED BY THE FIVE-POINT FORMULA.  THE
C       RESULTING SYSTEM OF EQUATIONS IS SOLVED BY THE
C       GAUSS-SEIDEL METHOD.
C
        DIMENSION A(150,26)
        READ(5,1) M,N,ITER,ERR
      1 FORMAT(3I10,F10.5)
        M1=M+1
        M2=M+2
        N1=N+1
        N2=N+2
        NM=N2-M1
        BB=M1
C
C       INITIAL GUESS IS ZERO.
C
        A(1,1)=0.
        DO 2 J=1,M
        A(1,J+1)=J/BB
      2 CONTINUE
        A(1,M2)=1.
        ME=M1
        DO 4 I=2,N2
        DO 3 J=1,ME
        A(I,J)=0.
      3 CONTINUE
        IF(I.LE.NM) A(I,M2)=1.
        IF(I.GT.NM)ME=ME-1
      4 CONTINUE
C
C       GAUSS-SEIDEL ITERATION.
C
        NC=0
      5 ERROR=0.
        ME=N1
        NC=NC+1
        DO 9 J=2,M1
        ME=ME-1
        DO 7 I=2,ME
        TEMP=0.25*(A(I,J-1)+A(I,J+1)+A(I-1,J)+A(I+1,J))
        ERROR=AMAX1(ERROR,ABS (TEMP-A(I,J)) )
      6 A(I,J)=TEMP
      7 CONTINUE
        ME1=ME+1
        TEMP=0.5*(A(ME1,J-1)+A(ME,J))
        ERROR=AMAX1(ERROR,ABS (TEMP-A(ME1.J)) )
      8 A(ME1,J)=TEMP
      9 CONTINUE
        IF(ERROR.LE.ERR) GOTO 11
```

```
      IF(NC.LT.ITER) GOTO 5
      WRITE(6,10)
   10 FORMAT(' NO CONVERGENCE')
   11 WRITE(6,12) NC,ERROR
   12 FORMAT(I10,E20.5)
      ME=N2
      DO 14 I=1,N2
      WRITE(6,13)(A(I,J),J=1,ME)
   13 FORMAT(1X,13F10.5)
      IF(I.GE.NM) ME=ME-1
   14 CONTINUE
      STOP
      END
```

```
C      GAUSSIAN-ELIMINATION IS IMPLEMENTED TO FIND
C      THE SOLUTION TO LAPLACE'S EQUATION IN AN
C      L-SHAPED CHANNEL.  LAPLACE'S EQUATION IS
C      APPROXIMATED BY THE FIVE-POINT FORMULA.
C
       DIMENSION A(540,45),B(540)
       COMMON A,B
       READ(5,1) M,N
     1 FORMAT(2I10)
       M1=M-1
       M2=M-2
       J1=M1*N-M1*M2/2
       J2=J1+1
       J4=J2+N-M
       J3=J4-1
C
C      GENERATION OF THE RIGHT-HAND SIDE B.
C
       DO 2 J=1,J1
       B(J)=0.
     2 CONTINUE
       J=-N
       BB=M+1
       DO 3 K=1,M
       J=J+N-K+2
       B(J)=-K/BB
     3 CONTINUE
       B(J)=B(J)-1.
       JJ=J+1
       DO 4 K=JJ,J3
       B(K)=-1.
     4 CONTINUE
       B(J4)=0.
C
C      GENERATION OF THE FIRST BLOCK
C
       DO 6 I=1,N
       DO 5 J=1,N
       A(I,J)=0.
     5 CONTINUE
       A(I,I)=-4.
       IF(I.NE.N) A(I,I+1)=1.
       IF(I.NE.1) A(I,I-1)=1.
     6 CONTINUE
       A(N,N-1)=2.
C
C      DECOMPOSITION OF THE FIRST M-1 BLOCKS.
C
       DO 7 K=1,M1
       CALL DECOM5(K,N)
```

```
    7 CONTINUE
C
C     DECOMPOSITION OF THE MTH BLOCK.
C
      NM=N-M1
      NE=NM
      DO 10 I=J2,J3
      I1=I+1
      TP=-1./A(I,1)
      DO 9J=I1,J4
      TEMP=TP*A(J,1)
      DO 8 K=2,NE
      A(J,K-1)=A(J,K)+TEMP*A(I,K)
    8 CONTINUE
      B(J)=B(J)+TEMP*B(I)
    9 CONTINUE
      NE=NE-1
   10 CONTINUE
C
C     BACK-SUBSTITUTION ON THE MTH BLOCK.
C
      A(J4,1)=B(J4)/A(J4,1)
      J3=J4+1
      DO 12 L=2,NM
      I=J3-L
      TEMP=B(I)
      DO 11 K=2,L
      TEMP=TEMP-A(I,K)*A(I+K-1,1)
   11 CONTINUE
      A(I,1)=TEMP/A(I,1)
   12 CONTINUE
C
C     BACK-SUBSTITUTION ON THE REMAINING BLOCKS.
C
      DO 15 L=1,M1
      J2=I
      NM=NM+1
      DO 14 J=1,NM
      I=J2-J
      TEMP=B(I)
      IF(J.NE.1) TEMP=TEMP-A(I+NM,1)
      DO 13 K=2,NM
      TEMP=TEMP-A(I,K)*A(I+K-1,1)
   13 CONTINUE
      A(I,1)=TEMP/A(I,1)
   14 CONTINUE
   15 CONTINUE
      J2=0
      NM=N
      DO 17 I=1,M
```

```
      NM=NM-1
      J1=J2+1
      J2=J1+NM
      WRITE(6,16)(A(J,1),J=J1,J2)
   16 FORMAT(1X,13F10.5)
   17 CONTINUE
      STOP
      END
```

```
      SUBROUTINE DECOM5(L,NT)
C
C     DECOMPOSITION OF THE CHANNEL PROBLEM BY
C     GAUSSIAN-ELIMINATION ON THE FIVE-POINT
C     FORMULA.  BLOCK K...BLOCK SIZE N.
C
      DIMENSION A(540,45),B(540)
      LOGICAL INEJ
      COMMON A,B
      N1=NT-L
      N=N1+1
      J2=L*NT-L*(L-1)/2
      J1=J2-N1
      J3=J2-1
      DO 4 I=J1,J2
      NI=N+I
      NI1=NI-1
      I1=I+1
      INEJ=I.NE.J2
      TP=-1./A(I,1)
      DO 2 J=I1,NI1
      TEMP=TP*A(J,1)
      DO 1 K=2,N
      A(J,K-1)=A(J,K)+TEMP*A(I,K)
    1 CONTINUE
      IF(INEJ)A(J,N)=TEMP
      B(J)=B(J)+TEMP*B(I)
    2 CONTINUE
      IF(.NOT.INEJ) GOTO 5
      IF(I.NE.J1) A(NI1,N)=A(NI1,N)+1.
      IF(I.EQ.J3) TP=2.*TP
      DO 3 K=2,N
      A(NI,K-1)=A(I,K)*TP
    3 CONTINUE
      IF(I.NE.J1) A(NI,N1)=A(NI,N1)+1.
      IF(I.EQ.J3) A(NI,N1)=A(NI,N1)+1.
      A(NI,N)=TP-4.
      B(NI)=B(NI)+TP*B(I)
    4 CONTINUE
    5 RETURN
      END
```

```
C       LAPLACE'S EQUATION IN AN L-SHAPED CHANNEL IS
C       APPROXIMATED BY THE NINE-POINT FORMULA.  THE
C       RESULTING SYSTEM OF EQUATIONS IS SOLVED BY THE
C       GAUSS-SEIDEL METHOD.
C
        DIMENSION A(150,26)
        READ(5,1) M,N,ITER,ERR
      1 FORMAT(3I10,F10.5)
        M1=M+1
        M2=M+2
        N1=N+1
        N2=N+2
        NM=N2-M1
        BB=M1
C
C        INITIAL GUESS IS ZERO.
C
        A(1,1)=0.
        DO 2 J=1,M
        A(1,J+1)=J/BB
      2 CONTINUE
        A(1,M2)=1.
        ME=M1
        DO 4 I=2,N2
        DO 3 J=1,ME
        A(I,J)=0.
      3 CONTINUE
        IF(I.LE.NM) A(I,M2)=1.
        IF(I.GT.NM)ME=ME-1
      4 CONTINUE
C
C       GAUSS-SEIDEL ITERATION.
C
        NC=0
      5 ERROR=0.
        ME=N1-1
        NC=NC+1
        DO 9 J=2,M1
        ME=ME-1
        DO 7 I=2,ME
        TEMP=0.05*(4.*(A(I,J-1)+A(I,J+1)+A(I-1,J)+A(I+1,J))
       1+A(I+1,J+1)+A(I+1,J-1)+A(I-1,J+1)+A(I-1,J-1))
        ERROR=AMAX1(ERROR,ABS((TEMP-A(I,J)))
        A(I,J)=TEMP
      7 CONTINUE
        ME1=ME+1
        ME2=ME+2
        TEMP=(4.*(A(ME1,J-1)+A(ME1,J+1)+A(ME,J)+A(ME2,J))
       1+A(ME2,J-1)+A(ME,J-1)+A(ME,J+1))/19.
        ERROR=AMAX1(ERROR,ABS((TEMP-A(ME1,J)))
```

```
      A(ME1,J)=TEMP
      TEMP=0.05*(8.*(A(ME2,J-1)+A(ME1,J))
     1+2.*A(ME1,J-1)+A(ME1,J+1)+A(ME+3,J-1))
      ERROR=AMAX1(ERROR,ABS((TEMP-A(ME2,J))))
      A(ME2,J)=TEMP
    9 CONTINUE
      IF(ERROR.LE.ERR) GOTO 11
      IF(NC.LT.ITER) GOTO 5
      WRITE(6,10)
   10 FORMAT(' NO CONVERGENCE')
      WRITE(6,12) NC,ERROR
   12 FORMAT(I10,E20.5)
      ME=M2
      DO 14 I=1,N2
      WRITE(6,13)(A(I,J),J=1,ME)
   13 FORMAT(1X,13F10.5)
      IF(I.GE.NM) ME=ME-1
   14 CONTINUE
      STOP
      END
```

```
C       LAPLACE'S EQUATION IN AN L-SHAPED CHANNEL
C       IS APPROXIMATED BY THE NINE-POINT FORMULA.
C       GAUSSIAN-ELIMINATION IS USED TO SOLVE THE
C       RESULTING SYSTEM OF EQUATIONS.
C
        DIMENSION A(540,45),B(540)
        COMMON /BLK/A,B
      1 FORMAT(2I10)
        M0=M-1
        N0=N-1
        NM=N-M+1
        J6=M*N-M0*M/2
        J5=J6-1
        J4=J6-2
        J1=J6-NM
        J2=J1+1
        J3=J1+2
C
C       THE RIGHT-HAND SIDE B IS GENERATED.
C
        DO 2 I=1,J1
        B(I)=0.
      2 CONTINUE
        BB=M+1
        I=-N
        DO 3 J=1,M0
        I=I+N-J+2
        B(I)=-6.*J/BB
      3 CONTINUE
        B(J2)=-(11.*M+5.)/BB
        DO 4 I=J3,J4
        B(I)=-6.
      4 CONTINUE
        B(J5)=-5.
        B(J6)=-1.
C
C       THE FIRST BLOCK IS GENERATED.
C
        DO 6 I=1,N
        DO 5 J=1,N
        A(I,J)=0.
      5 CONTINUE
        A(I,I)=-20.
        IF(I.NE.N) A(I,I+1)=4.
        IF(I.NE.1) A(I,I-1)=4.
      6 CONTINUE
        A(N0,N0)=-19.
        A(N,N0)=8.
C
C       THE FIRST M-1 BLOCKS ARE DECOMPOSED.
```

```
C
      DO 7 K=1,MP
      CALL DECOMP(K,M)
    7 CONTINUE
C
C     THE M'TH BLOCK IS DECOMPOSED.
C
      NE=NM
      DO 10 I=J2,J5
      I1=I+1
      TP=-1./A(I,1)
      DO 9 J=I1,J6
      TEMP=TP*A(J,1)
      DO 8 K=2,NE
      A(J,K-1)=A(J,K)+TEMP*A(I,K)
    8 CONTINUE
      B(J)=B(J)+TEMP*B(I)
    9 CONTINUE
      NE=NE-1
   10 CONTINUE
C
C     BACK-SUBSTITUTION ON THE M'TH BLOCK.
C
      B(J6)=B(J6)/A(J6,1)
      NM=NM-1
      DO 12 J=1,NM
      I=J6-J
      NE=NE+1
      TEMP=B(I)
      DO 11 K=2,NE
      TEMP=TEMP-A(I,K)*B(I+K-1)
   11 CONTINUE
      B(I)=TEMP/A(I,1)
   12 CONTINUE
C
C     BACK-SUBSTITUTION ON THE REMAINING M-1 BLOCKS.
C
      NM=NM+1
      DO 15 L=1,MP
      NM=NM+1
      NE=NM
      DO 14 J=1,NM
      I=J2-J
      TEMP=B(I)
      IF(J.GT.2) TEMP=TEMP-B(I+NE)
      IF(J.EQ.2) NE=NE+1
      DO 13 K=2,NE
      TEMP=TEMP-A(I,K)*B(I+K-1)
   13 CONTINUE
      B(I)=TEMP/A(I,1)
```

```
   14 CONTINUE
      J2=J2-NM
   15 CONTINUE
C
C     OUTPUT OF RESULTS.
C
      MM=M+2
      N2=N+2
      NM=N
      J1=0
      DO 16 J=1,N2
      A(1,J)=0.
   16 CONTINUE
      DO 18 I=1,M
      I1=I+1
      A(I1,1)=+I/23
      DO 17 K=1,NM
      A(I1,K+1)=B(J1+K)
   17 CONTINUE
      J1=J1+NM
      NM=NM-1
   18 CONTINUE
      NM=NM+1
      DO 19 K=1,NM
      A(MM,K)=1.
   19 CONTINUE
      DO 21 J=1,N2
      IF(J.GT.NM) MM=MM-1
      WRITE(6,20)(A(I,J),I=1,MM)
   20 FORMAT(1X,13F10.5)
   21 CONTINUE
      STOP
      END
```

```
      SUBROUTINE DECOM9(L,NT)
C
C     GAUSSIAN-ELIMINATION ROUTINE FOR DECOMPOSING
C     THE L'TH BLOCK OF THE L-SHAPED CHANNEL.
C     BLOCK SIZE IS N.  LAPLACE'S EQUATION IS
C     APPROXIMATED BY THE NINE-POINT FORMULA.
C
      DIMENSION A(540,45),B(540)
      LOGICAL ILTJ
      COMMON /BLK/A,B
      NO=NT-L
      N=NO+1
      N1=N+1
      J2=L*NT-L*(L-1)/2
      J1=J2-NO
      J3=J2-1
      J4=J2-2
      J5=J2-3
      J6=J1+1
C
C     ELIMINATION OF THE FIRST COLUMN.
C
      I1=J1+1
      NI=J2+1
      NI1=J2+2
      A(J1,N1)=4.
      TP=-1./A(J1,1)
      DO 2 J=I1,J2
      TEMP=TP*A(J,1)
      DO 1 K=2,N
      A(J,K-1)=A(J,K)+TEMP*A(J1,K)
    1 CONTINUE
      A(J,N)=4.*TEMP
      A(J,N1)=TEMP
      B(J)=B(J)+TEMP*B(J1)
    2 CONTINUE
      A(I1,N)=A(I1,N)+1.
      A(I1,N1)=A(I1,N1)+4.
      A(I1+1,N1)=A(I1+1,N1)+1.
      TEMP=4.*TP
      A(NI,1)=TEMP*A(J1,2)+1.
      DO 3 K=3,N
      A(NI,K-1)=TEMP*A(J1,K)
    3 CONTINUE
      A(NI,N)=4.*TEMP-20.
      A(NI,N1)=TEMP+4.
      B(NI)=B(NI)+TEMP*B(J1)
      A(NI1,1)=TP*A(J1,2)+4.
      A(NI1,2)=TP*A(J1,3)+1.
      DO 4 K=4,N
```

```
      A(NI1,K-1)=TP*A(J1,K)
    4 CONTINUE
      A(NI1,N)=TEMP+4.
      A(NI1,N1)=TP-20.
      B(NI1)=B(NI1)+TP*B(J1)
C
C     ELIMINATION OF REMAINING N-1 COLUMNS.
C
      DO 10 I=J5,J2
      I1=I+1
      TP=-1./A(I,1)
      IF(I.NE.J2) GOTO 5
      N1=N
      N=N-1
    5 NI=N+I
      NI1=NI+1
      ILTJ=I.LT.J3
      DO 7 J=I1,NI
      TEMP=TP*A(J,1)
      DO 6 K=2,N1
      A(J,K-1)=A(J,K)+TEMP*A(I,K)
    6 CONTINUE
      IF(ILTJ) A(J,N1)=TEMP
      B(J)=B(J)+TEMP*B(I)
    7 CONTINUE
      IF(.NOT.ILTJ) GOTO 10
      A(I1,N1)=A(I1,N1)+4.
      A(I+2,N1)=A(I+2,N1)+1.
      A(NI,N1)=A(NI,N1)+4.
      IF(I.EQ.J4) TP=2.*TP
      A(NI1,1)=TP*A(I,2)+4.
      A(NI1,2)=TP*A(I,3)+1.
      DO 8 K=4,N
      A(NI1,K-1)=TP*A(I,K)
    8 CONTINUE
      A(NI1,N)=TP*A(I,N1)+4.
      A(NI1,N1)=TP-20.
      IF(I.EQ.J5) A(NI1,N1)=A(NI1,N1)+1.
      IF(I.NE.J4) GOTO 9
      A(NI1,1)=A(NI1,1)+4.
      A(NI1,N)=A(NI1,N)+4.
    9 B(NI1)=B(NI1)+TP*B(I)
   10 CONTINUE
      RETURN
      END
```

APPENDIX IV

PROGRAM LISTING TO SOLVE THE CANAL PROBLEM IN THE PHYSICAL PLANE

A Fortran IV program listing to solve the Canal Problem 5.1 in the physical plane is included in this section. The method used is described in Section 5.2.

The subroutines SDPSI and SDPHI ascertain, by the use of the point-Gauss-Seidel method, the values of ψ and ϕ, respectively, at the gridpoints of a region which is determined by a particular choice of the free boundary. The subroutine PRINT may be used to print the values of ψ and ϕ. FREEX and FREEY determine the x-coordinate on the free boundary for a particular choice of Y, and the y-coordinate on the free boundary for a particular choice of X, respectively; while CR calculates the partial derivatives by use of differentiation formulae of $O(h^2)$ (see Appendix I), in order that the Cauchy-Riemann equations may be verified.

```
      REAL PHI(99,99),PSI(99,99)
      COMMON C1,C2 ,PSI,PHI,H,C
C
C     FREE BOUNDARY PROBLEM
C
   19 READ(5,100) PHIC,D,C,A,B,H,ERR
  100 FORMAT(7F10.0)
      C1=(B-A)/D**2
      C2=A
      K1=1+(D-C)/H +1.E-8
      K2=1-C/H +1.E-8
C
C     INITIAL GUESS
C
      J=1
      X=C
 1    IF(J.LE.K1) GOTO 3
      IF(J.LE.K2) GOTO 2
      IF(X.GT.C2+H/2)GOTO 8
      PSI0 = X/C2
      PSI0 = (PSI0-2)*PSI0 + 1
      PHI0=0.
      GOTO 4
 2    PSI0=1.
      PHI0=X*PHIC/D
      GOTO 4
 3    PSI0 = ((X-2*C)*X + C*C)/(D-C)**2
      PHI0=PHIC
 4    I=1
      YN = FREEY(X)
      PSI(1,J) = PSI0
      PHI(1,J) = PHI0
 5    I=I+1
      Y = (1-I)*H
      IF(Y.LT.YN)GOTO 6
      CONS = Y/YN
      PHI(I,J) = (1-CONS) * PHI0+Y
      PSI(I,J) =((2*CONS-3)*CONS*CONS+1)*PSI0
      GOTO 5
 6    DO 9K=I,20
      PSI(K,J) = 0
      PHI(K,J) = (1-K)*H
 9    CONTINUE
      J=J+1
      X = C + (J-1)*H
      GOTO 1
C
C     SOLUTION TO FREE BOUNDARY PROBLEM
C
 8    CALL SOPSI(ERR,K1,K2)
```

```
CALL SDPHI(ERR,K1,K2)
CALL PRINT
CALL CR(A)
GOTO 19
END
```

```
      SUBROUTINE SORPSI(ERR,K1,K2)
      REAL PSI(99,99), PHI(99,99)
      COMMON C1,C2,PSI,PHI,H,C
C
C   GAUSS-SEIDEL ITERATION FOR PSI
C
      T = C2-H/2
      M = 1-FREEY(C)/H + 1.E-8
      DO 9 K=1,100
      ER1=0
C
C    BOUNDARY CONDITIONS ON CA
C
      KK1 = K1 - 1
      DO 1 J = 2,KK1
      TEMP = .25*(PSI(1,J+1) + PSI(1,J-1) + 2*PSI(2,J))
      ER1 = AMAX1(ER1,(ABS((TEMP-PSI(1,J))/TEMP)))
      PSI(1,J) = TEMP
 1    CONTINUE
C
C    BOUNDARY CONDITIONS ON OA
C
      J=K2+1
      X=H
 2    TEMP = .25*(PSI(1,J+1) + PSI(1,J-1) + 2*PSI(2,J))
      ER1 = AMAX1(ER1,(ABS((TEMP-PSI(1,J))/TEMP)))
      PSI(1,J) = TEMP
      J=J+1
      X=X+H
      IF(X.LT.T ) GOTO 2
C
C    INTERIOR POINTS
C
      DO 8 I=2,M
      Y = (1-I)*H
      XN=FREEX(Y)
      N = K2 + XN/H-1
      IF(N.LT.2) GOTO 8
      DO 5 J=2,N
      X = C + (J-1)*H
      YL = (Y-FREEY(X))/H
      IF(YL.LT.1.) GOTO 3
      TEMP = .25*(PSI(I+1,J) + PSI(I-1,J) + PSI(I,J+1) + PSI(I,J-1))
      GOTO 4
 3    TEMP = ((YL-1)*PSI(I-2,J)/(YL+2) - 2*(YL-2)*PSI(I-1,J)/(YL+1)
     1+PSI(I,J+1) + PSI(I,J-1))*YL/(YL+3)
 4    ER1 = AMAX1(ER1,(ABS((TEMP-PSI(I,J))/TEMP)))
      PSI(I,J) = TEMP
 5    CONTINUE
      X = C+N*H
```

```
      XL = (XN-X)/H
      YL= (Y-FREEY(X))/H
      J = N+ 1
      IF(I.EQ.2) GOTO 12
      IF(YL.LT.1.) GOTO 6
12    TEMP = ((XL-1)*PSI(I,J-2)/(XL+2) - 2*(XL-2)*PSI(I,J-1)/(XL+1)
     1 + PSI(I+1,J) + PSI(I-1,J)) * XL/(XL+3)
      GOTO 7
6     TEMP = (2*(2-YL)*PSI(I-1,J)/(YL+1) + (YL-1)*PSI(I-2,J)/(YL+2)
     1 + 2*(2-XL)*PSI(I,J-1)/(XL+1) + (XL-1)*PSI(I,J-2)/(XL+2))
     2 *XL*YL/(XL*(3-YL)+YL*(3-XL))
7     ER1 = AMAX1(ER1,(ABS((TEMP-PSI(I,J))/TEMP)))
      PSI(I,J) = TEMP
8     CONTINUE
      IF(ERR.LT.ER1) GO TO 9
      WRITE(6,11) K,ER1
11    FORMAT('0 PSI  ITERATIONS', I5, '    ERROR',E15.5)
      RETURN
9     CONTINUE
      WRITE(6,10)ER1
10    FORMAT(' PSI DOES NOT CONVERGE ', F10.5)
      RETURN
      END
```

```
      SUBROUTINE SOPHI(ERR,K1,K2)
      REAL PSI(99,99), PHI(99,99)
      COMMON C1,C2,PSI,PHI,H,C
C
C   GAUSS-SEIDEL ITERATION FOR PHI
C
      M = 1-FREEY(C)/H
      DO 9 K=1,100
      ER1=0
C
C   BOUNDARY CONDITIONS ON DO
C
      J1=K1+1
      J2=K2-1
      DO 1 J=J1,J2
      TEMP = .25*(PHI(1,J+1) + PHI(1,J-1) + 2*PHI(2,J))
      ER1 = AMAX1(ER1,(ABS((TEMP-PHI(1,J))/TEMP)))
      PHI(1,J) = TEMP
 1    CONTINUE
      DO 8 I=2,M
      Y = (1-I)*H
C
C   BOUNDARY CONDITIONS ON EC
C
      IF(I.NE.M)GOTO 2
      YN = FREEY(C)
      YL = (Y-YN)/H
      IF(ABS(YL).LT.5.E-5)GOTO 3
      TEMP = ((YL-1)*PHI(I-2,1)/(YL+2)-2*(YL-2)*PHI(I-1,1)/(YL+1)
     1 + 2*PHI(I,2))*YL/(YL+3) + 6*YN/((YL+1)*(YL+2)*(YL+3))
      GOTO 13
 2    TEMP = .25*(PHI(I-1,1) + PHI(I+1,1) + 2*PHI(I,2))
 13   ER1 = AMAX1(ER1,(ABS((TEMP-PHI(I,1))/TEMP)))
      PHI(I,1) = TEMP
C
C   INTERIOR POINTS
C
      XN = FREEX(Y)
      N= K2 + XN/H - 1
      IF(N.LT.2) GOTO 8
      DO 5 J=2,N
      X = C + (J-1)*H
      YN = FREEY(X)
      YL = (Y-YN)/H
      IF(YL.LT.1.) GOTO 3
      TEMP = .25*(PHI(I+1,J) + PHI(I-1,J) + PHI(I,J+1) + PHI(I,J-1))
      GOTO 4
 3    TEMP = ((YL-1)*PHI(I-2,J)/(YL+2) - 2*(YL-2)*PHI(I-1,J)/(YL+1)
     1+PHI(I,J+1)+PHI(I,J-1))*YL/(YL+3)+6*YN/((YL+1)*(YL+2)*(YL+3))
 4    ER1 = AMAX1(ER1,(ABS((TEMP-PHI(I,J))/TEMP)))
```

```
      PHI(I,J) = TEMP
5     CONTINUE
      J = N+ 1
      X = C+M*H
      YN = FREEY(X)
      YL = (Y-YN)/H
      XL=(XN-X)/H
      IF(I.EQ.2) GOTO 12
      IF(YL.LT.1.) GOTO 6
12    TEMP = ((XL-1)*PHI(I,J-2)/(XL+2) - 2*(XL-2)*PHI(I,J-1)/(XL+1)
     1 + PHI(I+1,J)+PHI(I-1,J) + 6*Y /XL/(XL+1)/(XL+2))*XL/(XL+3)
      GOTO 7
6     TEMP = ((YL-1)*PHI(I-2,J)/(YL+2) + 2*(2-YL)*PHI(I-1,J)/(YL+1)
     1 + (XL-1)*PHI(I,J-2)/(XL+2) + 2*(2-XL)*PHI(I,J-1)/(XL+1)
     2 + 6*(YN/((YL+1)*(YL+2)*YL) + Y/((XL+1)*(XL+2)*XL)))
     3 *XL*YL/(XL*(3-YL)+YL*(3-XL))
7     ER1 = AMAX1(ER1,(ABS((TEMP-PHI(I,J))/TEMP)))
      PHI(I,J) = TEMP
8     CONTINUE
      IF(ERR.LT.ER1) GO TO 9
      WRITE(6,11) K,ER1
11    FORMAT('0 PHI  ITERATIONS', I5, '    ERROR',E15.5)
      RETURN
9     CONTINUE
      WRITE(6,10)ER1
10    FORMAT(' PHI DOES NOT CONVERGE ', F10.6)
      RETURN
      END
```

```
      SUBROUTINE PRINT
      REAL PSI(99,99), PHI(99,99) ,X(99)
      COMMON C1,C2,PSI,PHI,H,C
      X(1) = C
      DO 20 J = 2,99
      X(J) = X(J-1) + H
20    CONTINUE
      DO 3 K=1,99,12
      T = C+H*(K-1)
      IF(T.GT.C2) RETURN
      M = 1-FREEY(T)/H + 1.E-8
      WRITE(6,100) C1,C2
100   FORMAT('1     FREE BOUNDARY X = C1 * Y**2 +C2, C1 = ',F10.6,
     1' C2 = ' ,F10.6)
      WRITE(6,101)
101   FORMAT('0     PSI TABLE' )
      Y = -H
      DO 1 I=1,M
      Y=Y+H
      N=1+(FREEX(Y)-C)/H + 1.E-8
      IF(N.GT.K+11)N=K+11
      IF(I.EQ.1) WRITE(6,104) (X(J),J = K,N)
104   FORMAT(3X,12(4X,F6.3)/)
      WRITE(6,102)(PSI(I,J),J=K,N)
1     CONTINUE
      IF(M.GT.25) GOTO 14
      WRITE(6,103)
103   FORMAT('7     PHI TABLE' )
      GO TO 15
14    WRITE(6,100) C1,C2
      WRITE(6,105)
105   FORMAT('0     PHI TABLE')
15    Y = -H
      DO 2 I=1,M
      Y=Y+H
      N=1+(FREEX(Y)-C)/H + 1.E-8
      IF(N.GT.K+11)N=K+11
      IF(I.EQ.1) WRITE(6,104) (X(J),J = K,N)
      WRITE(6,102)(PHI(I,J),J=K,N)
102   FORMAT(5X,12F10.6)
2     CONTINUE
 3    CONTINUE
      RETURN
      END
```

```
      SUBROUTINE CR(A)
      DIMENSION PHI(99,99),PSI(99,99)
      COMMON C1,C2,PSI,PHI,H,C
      WRITE(6,7)
7     FORMAT('0                DPHI/DX                    DPSI/DY                    DPHI/D
     1Y               DPSI/DX')
      N=(A-C)/H
      H2=2.*H
      DO 1 J=2,N
      I=-FREEY(C+H*(J-1))/H
      DPIX=(PHI(I,J+1)-PHI(I,J-1))/H2
      DSIY=(PSI(I-1,J)-PSI(I+1,J))/H2
      DPIY=(PHI(I-1,J)-PHI(I+1,J))/H2
      DSIX=(PSI(I,J+1)-PSI(I,J-1))/H2
      WRITE(6,8) DPIX,DSIY,DPIY,DSIX
8     FORMAT(4E20.4)
    1 CONTINUE
      RETURN
      END
```

```
      REAL FUNCTION FREEX(Y)
      REAL PHI(99,99),PSI(99,99)
      COMMON C1,C2 ,PSI,PHI,H,C
      FREEX=C1*Y*Y+C2
      RETURN
      END
```

```
      REAL FUNCTION FREEY(X)
      REAL PHI(99,99),PSI(99,99)
      COMMON C1,C2 ,PSI,PHI,H,C
      FREEY = -SQRT(ABS((X-C2)/C1))
      RETURN
      END
```

APPENDIX V

PROGRAM LISTINGS TO SOLVE THE CANAL PROBLEM IN THE COMPLEX-POTENTIAL PLANE

The following pages contain Fortran IV program listings to solve, by the point-Gauss-Seidel and the Gaussian elimination methods, the systems resulting when Laplace's equation for the Canal Problem 5.2 in the complex-potential plane is approximated by the five-point and the nine-point formulae. The procedure that is used in each case is described in Section 5.3, and determines the y-coordinates at the gridpoints of the region. The subroutines X5 and X9, which use numerical differentiation and integration formulae of $O(h^4)$ and $O(h^6)$, respectively (see Appendix I), can be used to determine the x-coordinates at the gridpoints.

A listing of Schechter's method for the five-point formula is also included.

```
C       SOLUTION OF THE TRANSFORMED CANAL PROBLEM
C       BY THE GAUSS-SEIDEL METHOD.  LAPLACE'S EQUATION
C       IS APPROXIMATED BY THE FIVE-POINT FORMULA.
C       THE MIRROR IMAGE IS IMPLEMENTED ALONG THE
C       BOUNDARY WITH THE NORMAL DERIVATIVE SPECIFIED.
C
        REAL X(11,41),Y(11,41)
        COMMON X,Y
        READ(5,1) PHIA,PHIB,PHIC,H,ERR,ITER
    1   FORMAT(5F10.0,I10)
        N=1.E-4-PHIC/H
        K1=-1.E-4+1.+(PHIB-PHIC)/H
        K2=1.E-4+2.+(PHIA-PHIC)/H
        M=1.E-4+1./H
        NN=N+1
        KK=K2-K1-1
        ID=N-KK
        IF(K2.GT.N) ID=ID-1
        MM=M+1
C
C       INITIAL GUESS IS ZERO.
C
        DO 3 I=1,MM
        DO 2 J=1,NN
        Y(I,J)=0.0
    2   CONTINUE
    3   CONTINUE
C
        DO 4 J=1,KK
        K=K1+J
        Y(MM,K)=(K-NN)*H
    4   CONTINUE
C
C       GAUSS-SEIDEL ITERATION ON INTERIOR POINTS.
C
        NC=0
    5   ERROR=0.
        NC=NC+1
        DO 7 I=2,M
        DO 6 J=2,N
        TEMP=0.25*(Y(I+1,J)+Y(I-1,J)+Y(I,J+1)+Y(I,J-1))
        ERROR=AMAX1(ERROR,ABS(TEMP-Y(I,J)))
        Y(I,J)=TEMP
    6   CONTINUE
    7   CONTINUE
C
C       GAUSS-SEIDEL ITERATION ON BDDY. POINTS.
C
        DO 8 J=2,ID
        K=J
```

```
      IF(K.GT.K1)K=K+KK
      TEMP=0.25* (Y(MM,K-1)+Y(MM,K+1)+2.*Y(MM,K))
      ERROR=AMAX1(ERROR,ABS(TEMP-Y(MM,K)))
      Y(MM,K)=TEMP
  8   CONTINUE
      IF(ERROR.LE.ERR) GO TO 10
      IF(NC.LT.ITER)GO TO 5
      WRITE(6,9)
 9    FORMAT(' NO CONVERGENCE')
 10   WRITE(6,11) ERROR, NC
  11  FORMAT(E20.6, I15)
      WRITE(6,12) ((Y(I,J),I=1,MM),J=1,NN)
 12   FORMAT(1X,11F10.5)
      STOP
      END
```

```
C       SOLUTION TO THE TRANSFORMED CANAL PROBLEM
C       BY GAUSSIAN ELIMINATION. LAPLACE'S EQUATION
C       IS APPROXIMATED BY THE FIVE-POINT FORMULA. THE
C       MIRROR IMAGE IS IMPLEMENTED ALONG THE BOUNDARY
C       WITH THE NORMAL DERIVATIVE SPECIFIED.
C       THE UPPER, TRI-DIAGONAL BAND OF WIDTH N+1
C       AFTER GAUSSIAN ELIMINATION IS STORED IN
C       A(N*(M+1),N). NO PIVOTING IS PERFORMED UNTIL
C       THE (M+1) ST BLOCK. THE RESULT AFTER BACK-
C       SUBSTITUTION IS STORED IN A(N*(M+1),1).
C
        REAL A(400,40),B(80)
        LOGICAL LOGL,LOGK
        COMMON A,B
        READ(5,100) PHIA,PHIB,PHIC,H
    100 FORMAT(4F10.0)
        N=1.E-5-(1.+PHIC/H)
        K1=-1.E-4+(PHIB-PHIC)/H
        K2=1.E-5+1.+(PHIA-PHIC)/H
        M=1.E-5-1.+1./H
        N0=N-1
        N1=N+1
        K11=K1+1
        KK=K2-K11
        ID=N-KK
        M0=M-1
        M2=M-2
        MN=M*N
        MN1=MN+1
        NJ=MN
C
C       CALCULATION OF FIRST UPPER-TRIANGULAR BLOCK.
C       BAND WIDTH IS N+1
C
        A(1,1)=-4.
        A(2,1)=-3.75
        A(2,N)=0.25
        A(N1,1)=0.25
        A(N1,N)=-3.75
        DO 6 I=2,N
        I1=I+1
        I2=N-I+2
        NI=N+I
        I3=NI-1
        TP=-1./A(I,1)
        A(I-1,2)=1.
        IF(I.EQ.N) GO TO 2
        DO 1 K=I2,N
        A(I1,K-1)=TP*A(I,K)
      1 CONTINUE
```

```
      A(I1,1)=TP-4.
      A(I1,N)=TP
    2 DO 4 J=N1,I3
      TEMP=TP*A(J,1)
      A(J,1)=TEMP
      DO 3 K=I2,N
      A(J,K-1)=A(J,K)+TEMP*A(I,K)
    3 CONTINUE
      A(J,N)=TEMP
    4 CONTINUE
      A(I3,N)=A(I3,N)+1.
      A(NI,1)=TP
      DO 5 K=I2,N
      A(NI,K-1)=TP*A(I,K)
    5 CONTINUE
      A(NI,NC)=A(NI,NC)+1.
      A(NI,N)=TP-4.
    6 CONTINUE
C
C     CALCULATION OF NEXT M-2 UPPER-TRIANGULAR BLOCKS.
C
      DO 15 LL=1,M2
      ML=N*LL
      DO 14 L=1,N
      I=ML+L
      I1=I+1
      NI=N+I
      I2=NI-1
      TP=-1./A(I,1)
      DO 8 J=I1,I2
      TEMP=TP*A(J,1)
      DO 7 K=2,N
      A(J,K-1)=A(J,K)+TEMP*A(I,K)
    7 CONTINUE
      A(J,N)=TEMP
    8 CONTINUE
      IF(L.NE.1) A(I2,N)=1.+TEMP
      DO 9 K=2,NC
      A(NI,K-1)=TP*A(I,K)
    9 CONTINUE
      A(NI,NC)=1.+TP*A(I,N)
      IF(L.EQ.1) A(NI,NC)=A(NI,NC)-1.
      A(NI,N)=TP-4.
   14 CONTINUE
   15 CONTINUE
C
C     ELIMINATION OF THE MTH BLOCK.
C
      NE=N+ID
      DO 16 I=K11,NE
```

```
      B(I)=0.
      IF(I.LT.K2) B(I)= (N1-I)*H
   16 CONTINUE
      B(N+K1)=B(K11)
      B(N+K11)=B(K2-1)
      ML=ML+N
      ME=N
      DO 19 L=1,N
      I=ML+L
      I1=L+1
      IF(L.GT.K11.AND.L.LE.K2) ME=ME-1
      LOGL=L.LE.K1.OR.L.GE.K2
      LOGK=L.GT.K1
      TP=-1./A(I,1)
      NI=L+ME
      I2=NI-1
      DO 13 LL=I1,I2
      J=ML+LL
      TEMP=TP*A(J,1)
      DO 17 K=2,ME
      A(J,K-1)=A(J,K)+TEMP*A(I,K)
   17 CONTINUE
      IF(LOGL) A(J,ME)=TEMP
      IF(LOGK)B(LL)=B(LL)+TEMP*B(L)
   13 CONTINUE
      IF(.NOT.LOGL) GO TO 19
      IF(L.NE.1.AND.L.NE.K2)A(J,ME)=1.+TEMP
      TEMP=2.*TP
      IF(LOGK)B(NI)=B(NI)+TEMP*B(L)
      NI=I+ME
      DO 18 K=2,ME
      A(NI,K-1)=TEMP*A(I,K)
   18 CONTINUE
      IF(L.NE.1.AND.L.NE.K2) A(NI,ME-1)=A(NI,ME-1)+1.
      A(NI,ME)=TEMP-4.
   19 CONTINUE
C
C     GAUSSIAN ELIMINATION WITH PARTIAL PIVOTING
C     ON THE (M+1) ST BLOCK.
C
      CALL GAUSS(N,M,ID,1.E-40)
C
C     BACK SUBSTITUTION ON THE MTH BLOCK.
C
      NK=N-K1
      NK1=NK+1
      NK2=N-K2+1
      NK22=NK2+1
      DO 21 L=1,N
      ML=MN-L
```

```
      ML1=ML+1
      IF(L.GE.NK22.AND.L.LT.NK1) ME=ME+1
      IF(L.LE.NK2) TEMP=B(N1-L)-A(ML1+ME,1)
      IF(L.GE.NK22.AND.L.LE.NK) TEMP=B(N1-L)
      IF(L.GE.NK1) TEMP=-A(ML1+ME,1)
      DO 20 LL=2,ME
      TEMP=TEMP-A(ML1,LL)*A(ML+LL,1)
   20 CONTINUE
      A(ML1,1)=TEMP/A(ML1,1)
   21 CONTINUE
C
C     BACK SUBSTITUTION ON BLOCKS 2 THRU M-1.
C
      DO 24 LL=1,M2
      ML1=ML+1
      DO 23 L=1,N
      I=ML1-L
      TEMP=A(I+N,1)
      DO 22 K= 2,N
      TEMP=TEMP+A(I,K)*A(I+K-1,1)
   22 CONTINUE
      A(I,1)=-TEMP/A(I,1)
   23 CONTINUE
      ML=ML-N
   24 CONTINUE
C
C    BACK SUBSTITUTION ON 1ST BLOCK.
C
      DO 26 L=1,N0
      I=N1-L
      I1=I-1
      I2=N-I+2
      TEMP=A(N+I,1)
      DO 25 K=I2,N
      TEMP=TEMP+A(I,K)*A(K+I1,1)
   25 CONTINUE
      IF(L.NE.1) TEMP=TEMP+A(I+1,1)
      A(I,1)=-TEMP/A(I,1)
   26 CONTINUE
      A(1,1)=(A(N1,1)+A(2,1))/4.
      M1=M+1
      DO 27 I=1,M1
      I1=(I-1)*N+1
      I2=I*N
      WRITE(6,28)(A(J,1),J=I1,I2)
   27 CONTINUE
   28 FORMAT(1X,12F10.5)
      STOP
      END
```

```
C       SOLUTION OF THE TRANSFORMED CANAL PROBLEM
C       BY THE GAUSS-SEIDEL METHOD.  LAPLACE'S EQUATION
C       IS APPROXIMATED BY THE NINE-POINT FORMULA.
C       THE MIRROR IMAGE IS IMPLEMENTED ALONG THE
C       BOUNDARY WITH THE NORMAL DERIVATIVE SPECIFIED.
C
        DIMENSION X(11,41),Y(11,41)
        COMMON X,Y
        READ(5,1) PHIA,PHIB,PHIC,H,ERR,ITER
    1   FORMAT(5F10.0,I10)
        N=1.E-4-PHIC/H
        K1=-1.E-4+1.+(PHIB-PHIC)/H
        K2=1.E-4+2.+(PHIA-PHIC)/H
        M=1.E-4+1./H
        NN=N+1
        KK=K2-K1-1
        ID=N-KK
        IF(K2.GT.N) ID=ID-1
        MM=M+1
C
C       INITIAL GUESS IS ZERO.
C
        DO 3 I=1,MM
        DO 2 J=1,NN
        Y(I,J)=0.0
    2   CONTINUE
    3   CONTINUE
C
        DO 4 J=1,KK
        K=K1+J
        Y(MM,K)=(K-NN)*H
    4   CONTINUE
C
C       GAUSS-SEIDEL ITERATION ON INTERIOR POINTS.
C
        NC=0
    5   ERROR=0.
        NC=NC+1
        DO 7 I=2,M
        DO 6 J=2,N
        TEMP = 0.05*(4*(Y(I+1,J)+Y(I-1,J)+Y(I,J+1)+Y(I,J-1))
       1+Y(I+1,J+1)+Y(I+1,J-1)+Y(I-1,J+1)+Y(I-1,J-1))
        ERROR=AMAX1(ERROR,ABS(TEMP-Y(I,J)))
        Y(I,J)=TEMP
    6   CONTINUE
    7   CONTINUE
C
C       GAUSS-SEIDEL ITERATION ON BODY. POINTS.
C
        DO 8 J=2,ID
```

```
      K=J
      IF(K.GT.K1)K=K+KK
      TEMP=0.1*(2.*(Y(MM,K-1)+Y(MM,K+1))
     1+4.*Y(M,K)+Y(M,K-1)+Y(M,K+1))
      ERROR=AMAX1(ERROR,ABS(TEMP-Y(MM,K)))
      Y(MM,K)=TEMP
8     CONTINUE
      IF(ERROR.LE.ERR) GO TO 10
      IF(NC.LT.ITER)GO TO 5
      WRITE(6,9)
9     FORMAT('  NO CONVERGENCE')
10    WRITE(6,11) ERROR,NC
 11   FORMAT(E20.6, I15)
      WRITE(6,12) ((Y(I,J),I=1,MM),J=1,NN)
 12   FORMAT(1X,11F10.5)
      STOP
      END
```

```
C       SOLUTION OF THE TRANSFORMED CANAL PROBLEM
C       BY GAUSSIAN ELIMINATION. LAPLACE'S EQUATION
C       IS APPROXIMATED BY THE NINE-POINT FORMULA.
C       THE MIRROR IMAGE IS IMPLEMENTED ALONG THE
C       BOUNDARY WITH THE NORMAL DERIVATIVE. THE
C       REDUCED MATRIX IS A BAND MATRIX OF BAND
C       WIDTH N+2 STORED IN A(M*(N+1),N+1).
C       PIVOTING IS NOT NECESSARY UNTIL THE (M+1)ST
C       BLOCK. THE RESULT AFTER BACK SUBSTITUTION
C       IS STORED IN A(N*(M+1),1).
C
        DIMENSION A(440,41), B(80)
        LOGICAL LOOK
        COMMON A,B
        READ(5,100) PHIA,PHIB,PHIC,H
  100   FORMAT (4F10.0)
        N=1.E-5-(1.+PHIC/H)
        K1=-1.E-4+(PHIB-PHIC)/H
        K2=1.E-5+1.+(PHIA-PHIC)/H
        M=1.E-5-1.+1./H
        KK=K2-K1-1
        ID=N-KK
        MN=M*N
        M0=M-1
        N1=N+1
        MN1=MN+1
        N0=N-1
        MN0=MN-N
C
C       GENERATION OF THE FIRST BLOCK.
C
        DO 2 I=1,N
        DO 1 J=1,N
        A(I,J)=0.
    1   CONTINUE
        IF(I.NE.1) A(I,I-1)=4.
        A(I,I)=-20.
        IF(I.NE.N) A(I,I+1)=4.
    2   CONTINUE
C
C       ELIMINATION OF THE FIRST M-1 COLUMNS.
C
        DO 3 K=1,M0
        CALL REDUCE (K,N)
    3   CONTINUE
C
C       GENERATION OF THE RIGHT-HAND SIDE B.
C
        K11=K1+1
        K12=K1+2
```

```
      K22=K2-2
      BNK=N-K1
      K10=K1-1
      K21=K2+1
      NID=N+ID
      DO 4 I=1,K10
      B(I)=0.
    4 CONTINUE
      B(K1)=BNK*H
      B(K11)=(5.*BNK-1.)*H
      H6=6*H
      DO 5 I=K12,K22
      BNK=N1-I
      B(I)=BNK*H6
    5 CONTINUE
      K20=K2-1
      B(K20)=(5.*(N-K2)+11.)*H
      B(K2)=(N-K22)*H
      DO 6 I=K21,NID
      B(I)=0.
    6 CONTINUE
      B(N+K1)=2.*B(K1)
      B(N+K11)=2.*B(K2)
C
C     ELIMINATION OF THE FIRST COLUMN OF THE
C     MTH BLOCK.
C
      I=MN-N0
      NI=N+I
      NI0=NI-1
      NI1=NI+1
      I1=I+1
      TP=-1./A(I,1)
      A(I,N1)=4.
      DO 8 J=I1,NI0
      TEMP=TP*A(J,1)
      DO 7 K=2,N
      A(J,K-1)=A(J,K)+TEMP*A(I,K)
    7 CONTINUE
      A(J,N)=4.*TEMP
      A(J,N1)=TEMP
    8 CONTINUE
      A(I1,N)=A(I1,N)+1.
      A(I1,N1)=A(I1,N1)+4.
      A(I+2,N1)=A(I+2,N1)+1.
      TEMP=4.*TP
      A(NI,1)=TEMP*A(I,2)+1.
      DO 9 K=3,N
      A(NI,K-1)=TEMP*A(I,K)
    9 CONTINUE
```

```
      A(NI,N)=4.*TEMP-10.
      A(NI,N1)=TEMP+2.
      A(NI1,1)=TP*A(I,2)+4.
      A(NI1,2)=TP*A(I,3)+1.
      DO 10 K=4,N
      A(NI1,K-1)=TP*A(I,K)
   10 CONTINUE
      A(NI1,N)=TEMP+2.
      A(NI1,N1)=TP-10.
C
C     ELIMINATION OF REMAINING COLUMNS OF THE
C     MTH BLOCK.
C
      NE=N1
      DO 14 L=2,N
      LOGK=L.LT.K1.OR.(L.GE.K20.AND.L.NE.N)
      IF(L.GT.K1.AND.L.LE.K20)NE=NE-1
      I=MN0+L
      I1=I+1
      L1=L+1
      NL=NE+L-1
      NI=NE+I-1
      NI1=NI+1
      TP=-1./A(I,1)
      DO 12 LL=L1,NL
      J=MN0+LL
      TEMP=TP*A(J,1)
      DO 11 K=2,NE
      A(J,K-1)=A(J,K)+TEMP*A(I,K)
   11 CONTINUE
      IF(LOGK) A(J,NE)=TEMP
      B(LL)=B(LL)+TEMP*B(L)
   12 CONTINUE
      IF(.NOT.LOGK) GO TO 14
      A(I1,NE)=A(I1,NE)+4.
      IF(L.NE.N0)  A(I+2,NE)=A(I+2,NE)+1.
      IF(L.NE.K20)  A(NI,NE)=A(NI,NE)+2.
      A(NI1,1)=TP*A(I,2)+4.
      A(NI1,2)=TP*A(I,3)+1.
      IF(L.EQ.N0)  A(NI1,2)=A(NI1,2)-1.
      DO 13 K=4,NE
      A(NI1,K-1)=TP*A(I,K)
   13 CONTINUE
      IF(L.NE.K20)  A(NI1,NE-1)=A(NI1,NE-1)+2
      A(NI1,NE)=TP-10.
      B(NL+1)=B(NL+1)+TP*B(L)
   14 CONTINUE
C
C     GAUSSIAN ELIMINATION WITH PARTIAL PIVOTING
C     ON THE (M+1) ST BLOCK.
```

```
C
      CALL GAUSS(N,M,ID,1.E-40)
C
C     BACK SUBSTITUTION ON THE MTH BLOCK.
C
      DO 16 L=1,N
      I=MN1-L
      LOGK=L.GE.N-K2+3.AND.L.LE.N1-K1
      IF(LOGK)NE=NE+1
      TEMP=B(N1-L)
      IF(L.NE.1.AND..NOT.LOGK)TEMP=TEMP-A(I+NE,1)
      I1=I-1
      DO 15 J=2,NE
      TEMP=TEMP-A(I,J)*A(I1+J,1)
   15 CONTINUE
      A(I,1)=TEMP/A(I,1)
   16 CONTINUE
C
C     BACK SUBSTITUTION ON REMAINING M-1 BLOCKS.
C
      DO 19 K=1,M1
      MK=(M-K)*N+1
      DO 18 L=1,N
      I=MK-L
      I1=I-1
      TEMP=0.
      IF(L.NE.1) TEMP=A(I+N1,1)
      DO 17 J=2,N1
      TEMP=TEMP+A(I,J)*A(I1+J,1)
   17 CONTINUE
      A(I,1)=-TEMP/A(I,1)
   18 CONTINUE
   19 CONTINUE
      MID=MN+ID
      WRITE(6,20)(A(I,1),I=1,MID)
   20 FORMAT(1X,10E12.5)
      STOP
      END
```

```
      SUBROUTINE REDUCE(L,N)
C
C     DECOMPOSITION OF THE NINE-POINT FORMULA BY
C     GAUSSIAN-ELIMINATION WITHOUT PIVOTING.
C     BLOCK K...BLOCK SIZE N.
C
      DIMENSION A(440,41)
      LOGICAL IMEN
      COMMON A
      N0=N-1
      N1=N+1
C
C     ELIMINATION OF 1ST COLUMN.
C
      I=(L-1)*N+1
      I1=I+1
      NI=N+I
      KN=L*N
      NI1=NI+1
      A(I,N1)=4.
      TP=-1./A(I,1)
      DO 2 J=I1,KN
      TEMP=TP*A(J,1)
      DO 1 K=2,N
      A(J,K-1)=A(J,K)+TEMP*A(I,K)
    1 CONTINUE
      A(J,N)=4.*TEMP
      A(J,N1)=TEMP
    2 CONTINUE
      A(I1,N)=A(I1,N)+1.
      A(I1,N1)=A(I1,N1)+4.
      A(I+2,N1)=A(I+2,N1)+1.
      TEMP=4.*TP
      A(NI,1)=TEMP*A(I,2)+1.
      DO 3 K=3,N
      A(NI,K-1)=TEMP*A(I,K)
    3 CONTINUE
      A(NI,N)=4.*TEMP-20.
      A(NI,N1)=TEMP+4.
      A(NI1,1)=TP*A(I,2)+4.
      A(NI1,2)=TP*A(I,3)+1.
      DO 4 K=4,N
      A(NI1,K-1)=TP*A(I,K)
    4 CONTINUE
      A(NI1,N)=TEMP+4.
      A(NI1,N1)=TP-20.
C
C     ELIMINATION OF REMAINING COLUMNS.
C
      KN0=KN-1
```

```
      DO 8 I=I1,KN
      I1=I+1
      NI=N+I
      NI1=NI+1
      INEN=I.NE.KN
      TP=-1./A(I,1)
      DO 6 J=I1,NI
      TEMP=TP*A(J,1)
      DO 5 K=2,N1
      A(J,K-1)=A(J,K)+TEMP*A(I,K)
    5 CONTINUE
      IF(INEN)A(J,N1)=TEMP
    6 CONTINUE
      IF(.NOT.INEN) GOTO 9
      A(I1,N1)=A(I1,N1)+4.
      IF(I.NE.KN2) A(I+2,N1)= A(I+2,N1)+1.
      A(NI,N1)=A(NI,N1)+4.
      A(NI1,1)=TP*A(I,2)+4.
      DO 7 K=3,N1
      A(NI1,K-1)=TP*A(I,K)
    7 CONTINUE
      IF(I.NE.KN2)A(NI1,2)=A(NI1,2)+1.
      A(NI1,N)=A(NI1,N)+4.
      A(NI1,N1)=TP-20.
    8 CONTINUE
    9 RETURN
      END
```

```
      SUBROUTINE GAUSS(ID,M,N,CRIT)
C     GAUSSIAN ELIMINATION WITH PARTIAL PIVOTING ON AR=U.
C     RESULT STORED IN R. ROWS NOT PHYSICALLY INTERCHANGED.
C
      DIMENSION A(440,41),U(80)
      INTEGER INDEX(40)
      COMMON A,U
      MN=M*ID
      N1 = N-1
      N2 = N+1
      N3=ID+N+1
      MNN=MN+N
      MN1=MNN+1
      DO 1 I=1,N
      INDEX(I) = I
 1    CONTINUE
      DO 5 LL=1,N1
      I=MN+LL
      II=LL+1
      XAM = 0.
      DO 2 J=LL,N
      IN = INDEX(J)
      IF(XAM.GE.ABS(A(I,IN))) GO TO 2
      XAM = ABS(A(I,IN))
      K = J
 2    CONTINUE
      IF(XAM.LE.CRIT) GOTO 8
      KK = INDEX(K)
      MK=MN+KK
      NK=ID+KK
      INDEX(K)=INDEX(LL)
      INDEX(LL)=KK
      TP=-1./A(MK,LL)
      DO 4 L=II,N
      K = INDEX(L)
      ML=MN+K
      NL=ID+K
      TEMP=TP*A(ML,LL)
      DO 3 J=II,N
      A(ML,J)=A(ML,J)+TEMP*A(MK,J)
 3    CONTINUE
      U(NL)=U(NL)+TEMP*U(NK)
 4    CONTINUE
 5    CONTINUE
      K = INDEX(N)
      MK = MN+K
      IF(ABS(A(MK,N)).LE.CRIT) GOTO 8
      NK=ID+K
      A(MNN,1)=U(NK)/A(MK,N)
      DO 7 I=1,N1
```

```
      KK = N-I
      K = INDEX(KK)
      MK=MN+K
      NK=ID+K
      TM=U(NK)
      DO 6 J=1,I
      TM=TM-A(MK,N2-J)*A(MN1-J,1)
6     CONTINUE
      A(MK,1)=TM/A(MK,KK)
7     CONTINUE
      RETURN
8     WRITE(6,9)
9     FORMAT (' MATRIX IS SINGULAR')
      STOP
      END
```

```
C       SOLUTION TO THE CANAL PROBLEM
C       IN THE COMPLEX-POTENTIAL PLANE.
C       LAPLACE'S EQUATION IS APPROXIMATED
C       BY THE FIVE-POINT FORMULA.  SCHECHTER'S
C       METHOD IS USED TO SOLVE THE RESULTING
C       SYSTEM OF EQUATION.
C
        REAL MA(100,100),MB(100,100),MI(100,100),P(100,100)
        REAL B(100),C(100)
        READ(5,100)PHIA,PHIB,PHIC,H
  100   FORMAT(4F10.0)
        N=1.E-5-(1.+PHIC/H)
        K1=-1.E-4+(PHIB-PHIC)/H
        K2=1.E-5+1.+(PHIA-PHIC)/H
        M=1.E-5-1.+1./H
        K11=K1+1
        KK=K2-K11
        ID=N-KK
        M2=M-2
        N0=N-1
        N1=N+1
        M1=M+1
        M0=M-1
C
C       CALCULATION OF P(1) AND P(2).
C
        DO 2 I=1,N
        DO 1 J=1,N
        MA(I,J)=0.
        MB(I,J)=0.
    1   CONTINUE
        MA(I,I)=-4.
        MB(I,I)=17.
        IF(I.NE.1) MA(I,I-1)=1.
        IF(I.NE.N) MA(I,I+1)=1.
        IF(I.NE.1) MB(I,I-1)=-8.
        IF(I.GT.2) MB(I,I-2)=1.
        IF(I.NE.N) MB(I,I+1)=-8.
        IF(I.LT.N0) MB(I,I+2)=1.
    2   CONTINUE
        MB(1,1)=16.
        MB(N,N)=16.
C
C       CALCULATION OF P(M-2) AND P(M-1) FROM
C       P(K)=P(K-1)*D-P(K-2).   MA=P(M-2), MB=P(M-1).
C
        IF(M.EQ.3) GOTO 140
        DO 14 K=3,M0
        JK = N - K
        I2=N-K+1
```

```
      IF(K.EQ.2*(K/2)) GOTO 9
      DO 3 I=1,K
      MA(I,1)=-4.*MB(I,1)+MB(I,2)-MA(I,1)
 3    CONTINUE
      DO 4 I=I2,N
      MA(I,N)=MB(I,N0)-4.*MB(I,N)-MA(I,N)
 4    CONTINUE
      DO 6 J=2,N0
      J0=J-1
      J1=J+1
      I1=J0-K
      IF(I1.LT.1) I1=1
      I2=J0+K
      IF(I2.GT.N)I2=N
      DO 5 I=I1,I2
      MA(I,J)=MB(I,J0)-4.*MB(I,J)+MB(I,J1)-MA(I,J)
 5    CONTINUE
 6    CONTINUE
      DO 80 I=1,JK
      MA(I+K,I)=1.
 80   CONTINUE
      MA(JK,N)=1.
      IF(K.NE.40)GOTO 14
      DO 8 I=1,N
      DO 7 J=1,N
      TEMP=MA(I,J)
      MA(I,J)=MB(I,J)
      MB(I,J)=TEMP
 7    CONTINUE
 8    CONTINUE
      GOTO 14
 9    DO 10 I=1,K
      MB(I,1)=-4.*MA(I,1)+MA(I,2)-MB(I,1)
 10   CONTINUE
      DO 11 I=I2,N
      MB(I,N)=MA(I,N0)-4.*MA(I,N)-MB(I,N)
 11   CONTINUE
      DO 13 J=2,N0
      J0=J-1
      J1=J+1
      I1=J0-K
      IF(I1.LT.1) I1=1
      I2=J0+K
      IF(I2.GT.N)I2=N
      DO 12 I=I1,I2
      MB(I,J)=MA(I,J0)-4.*MA(I,J)+MA(I,J1)-MB(I,J)
 12   CONTINUE
 13   CONTINUE
      DO 130 I=1,JK
      MB(I+K,I)=1.
```

```
  130 CONTINUE
      MB(JK,N)=1.
  14  CONTINUE
C
C     MI=P(M-1) INVERSE.
C
  140 CALL INVERT(N,MB,MI)
C
C     MI=U(M-1) INVERSE.
C
      CALL MULT(N,N,N,MI,MA)
C
C     MB=U(M).
C
      DO 18 I=1,N
      I1=I+1
      DO 17 J=1,N
      MB(I,J)=-MI(I,J)
  17  CONTINUE
      MB(I,I)=MB(I,I)-4.
      IF(I.NE.1)MB(I,I-1)=MB(I,I-1)+1.
      IF(I.NE.N)MB(I,I+1)=MB(I,I+1)+1.
  18  CONTINUE
C
C     MA=U(M) INVERSE.
C
      CALL INVERT(N,MB,MA)
C
C     MB=L(M+1)
C
      DO 20 I=1,ID
      K=I
      IF(K.GT.K1)K=K+KK
      DO 19 J=1,N
      MB(I,J)=2.*MA(K,J)
  19  CONTINUE
  20  CONTINUE
C
C     GENERATE THE RIGHT-HAND SIDE B(1)
C     C=B(2)-L(M+1)*B(1)
C
      DO 21 I=1,KK
      K=I+K1
      B(K)=(N1-K)*H
  21  CONTINUE
      DO 23 I=1,ID
      TEMP=0.
      DO 22 J=1,KK
      K=K1+J
      TEMP=TEMP+MB(I,K)*B(K)
```

```
   22 CONTINUE
      C(I)=-TEMP
   23 CONTINUE
      C(K1)=C(K1)+B(K11)
      C(K11)=C(K11)+B(K2-1)
C
C     CALCULATE MB=U(M+1).
C
      DO 25 J=1,ID
      K=J
      IF(K.GT.K1)K=K+KK
      DO 24 I=1,ID
      MB(I,J)=-MA(I,K)
   24 CONTINUE
      MB(J,J)=MB(J,J)-4.
      IF(J.NE.1.AND.J.NE.K11) MB(J-1,J)=MB(J-1,J)+1.
      IF(J.NE.K1.AND.J.NE.ID) MB(J+1,J)=MB(J+1,J)+1.
   25 CONTINUE
C
C     SOLVE MB*B=C FOR B.
C
      CALL DECOMP(ID,MB)
      CALL SOLVE(ID,MB,C,B)
C
C     MB(M+1, )=B.
C
      DO 29 I=1,ID
      K=I
      IF(K.GT.K1)K=K+KK
      MB(M1,K)=B(I)
   29 CONTINUE
      DO 30 I=1,KK
      K=I+K1
      MB(M1,K)=(K-N1)*H
   30 CONTINUE
C
C     MB(M,I)=Y(M)=MA*(R-J*Y(M+1)).
C
      DO 31 I=1,N
      C(I)=-MB(M1,I)
   31 CONTINUE
      DO 35 I=1,N
      TEMP=0.
      DO 34 J=1,N
      TEMP=TEMP+MA(I,J)*C(J)
   34 CONTINUE
      MB(M,I)=TEMP
   35 CONTINUE
C
C     MB(M-1,)=Y(M-1)=MA*Y(M)
```

```
C
      DO 37 I=1,N
      TEMP=0.
      DO 36 J=1,N
      TEMP=TEMP+MI(I,J)*MB(M,J)
   36 CONTINUE
      MB(MO,I)=-TEMP
   37 CONTINUE
C
C     CALCULATION OF REMAINING Y(K),
C     Y(K)=-D*Y(K+1)-Y(K+2).
C
      DO 39 K=1,M2
      I1=M-K
      I=I1-1
      I2=I+2
      MB(I,1)=4.*MB(I1,1)-MB(I1,2)-MB(I2,1)
      MB(I,N)=4.*MB(I1,N)-MB(I1,NO)-MB(I2,N)
      DO 38 J=2,NO
      MB(I,J)=4.*MB(I1,J)-MB(I1,J+1)-MB(I1,J-1)-MB(I2,J)
   38 CONTINUE
   39 CONTINUE
      CALL CS0198(NMT)
      MMT=(NMT-MMT)
      RMT=MMT*26.E-6
      WRITE(6,200) RMT
      DO 40 I=1,M
      WRITE(6,200)(MB(I,J),J=1,N)
  200 FORMAT(1X,10F12.5)
   40 CONTINUE
      GOTO 999
      STOP
      END
```

```
      SUBROUTINE MULT(M,N,P,A,B)
C
C     WINOGRAD MULT'N OF A(M,N) AND B(N,P).
C     RESULT IN A.
C
      DOUBLE PRECISION ZETS(100),TEMP(100),ZETT,TP
      REAL A(100,100),B(100,100)
      INTEGER P
      N2=N/2
      DO 2 J=1,P
      TP=0.
      DO 1 I=1,N2
      K=2*I
      TP=TP+DBLE(B(K-1,J))*DBLE(B(K,J))
 1    CONTINUE
      ZETS(J)=TP
 2    CONTINUE
      ZETT=0.
      DO 3 J=1,N2
      K=2*J
      ZETT=ZETT+DBLE(A(I,K-1))*DBLE(A(I,K))
 3    CONTINUE
      DO 5 J=1,P
      TP=0.
      DO 4 L=1,N2
      K=2*L
      TP=TP+(DBLE(A(I,K-1))+DBLE(B(K,J)))
     1*(DBLE(A(I,K))+DBLE(B(K-1,J)))
 4    CONTINUE
      TEMP(J)=TP-ZETT-ZETS(J)
 5    CONTINUE
      IF(N.EQ.2*N2) GOTO 7
      TP=A(I,N)
      DO 6 J=1,P
      A(I,J)=TEMP(J)+TP*DBLE(B(N,J))
 6    CONTINUE
      GO TO 9
 7    DO 8 J=1,P
      A(I,J)=TEMP(J)
 8    CONTINUE
 9    CONTINUE
      RETURN
      END
```

```
      SUBROUTINE DECOMP(N,A)
C
C     DECOMPOSITION OF A INTO THE PRODUCT
C     OF A LOWER AND AN UPPER TRIANGULAR MATRIX..
C
      DIMENSION A(100,100),SCALES(100),IPS(100)
      COMMON IPS
      DO 5 I=1,N
      IPS(I)=I
      ROWNRM=0.0
      DO 2 J=1,N
    2 ROWNRM=AMAX1(ROWNRM,ABS(A (I,J)))
      IF(ROWNRM)3,4,3
    3 SCALES(I)=1.0/ROWNRM
      GO TO 5
    4 CALL SING(1)
      RETURN
    5 CONTINUE
      NM1=N-1
      DO 16 K=1,NM1
      BIG=0.0
      DO 11 I=K,N
      IP=IPS(I)
      SIZE=ABS(A (IP,K)*SCALES(IP))
      IF(SIZE-BIG)11,11,10
   10 BIG=SIZE
      IDXPIV=I
   11 CONTINUE
      IF(BIG)13,12,13
   12 CALL SING(2)
      RETURN
   13 IF(IDXPIV-K)14,15,14
   14 J=IPS(K)
      IPS(K)=IPS(IDXPIV)
      IPS(IDXPIV)=J
   15 KP=IPS(K)
      PIVOT=A (KP,K)
      KP1=K+1
      DO 16 I=KP1,N
      IP=IPS(I)
      EM=-A (IP,K)/PIVOT
      A(IP,K)=-EM
      DO 16 J=KP1,N
   16 A (IP,J)=A (IP,J)+EM*A (KP,J)
      KP=IPS(N)
   20 CONTINUE
      IF( A(KP,N))19,18,19
   18 CALL SING(2)
   19 RETURN
      END
```

```
      SUBROUTINE SOLVE(N,UL,B,X)
C
C     SOLUTION OF AX=B..UL IS THE
C     DECOMPOSED FORM OF A.
C
      DIMENSION UL(100,100),B(100),X(100),IPS(100)
      COMMON IPS
      NP1=N+1
      IP=IPS(1)
      X(1)=B(IP)
      DO 2 I=2,N
      IP=IPS(I)
      IM1=I-1
      SUM=0.0
      DO 1 J=1,IM1
    1 SUM=SUM+UL(IP,J)*X(J)
    2 X(I)=B(IP)-SUM
      IP=IPS(N)
      X(N)=X(N)/UL(IP,N)
      DO 4 IBACK=2,N
      I=NP1-IBACK
      IP=IPS(I)
      IP1=I+1
      SUM=0.0
      DO 3 J=IP1,N
    3 SUM=SUM+UL(IP,J)*X(J)
    4 X(I)=(X(I)-SUM)/UL(IP,I)
      RETURN
      END
```

```
      SUBROUTINE INVERT(N,A,AINV)
C
C     THE INVERSE OF A IS STORED IN AINV.
C
      DIMENSION A(100,100),AINV(100,100),B(100),X(100)
      CALL DECOMP(N,A)
      DO 3 J=1,N
      DO 1 I=1,N
      B(I)=0.0
      IF(I.EQ.J)B(I)=1.0
    1 CONTINUE
      CALL SOLVE(N,A ,B,X)
      DO 2 I=1,N
    2 AINV(I,J)=X(I)
    3 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE SING(IWHY)
      GOTO (1,2), IWHY
    1 WRITE(6,11)
      RETURN
    2 WRITE(6,12)
      RETURN
   11 FORMAT('0 MATRIX WITH ZERO ROW IN DECOMPOSE.')
   12 FORMAT('0 SINGULAR MATRIX IN DECOMPOSE. ZERO DIVIDE IN SOL
      END
```

```
      SUBROUTINE X9(MM,NN)
C
C     CALCULATION OF THE X-COORDINATES CORRESPONDING TO
C     THE GRID-POINTS OF THE CANAL PROBLEM IN THE
C     COMPLEX-POTENTIAL PLANE.  THE Y-COORDINATES HAVE
C     BEEN COMPUTED IMPLEMENTING THE NINE-POINT FORMULA.
C
      DIMENSION X(11,41),Y(11,41),YP(5),YS(5)
      LOGICAL JJ1,JJ2,JJ3,JM1,JM,JMM,JJG
      COMMON X,Y
      M=MM-1
      M1=M-1
      M2=M-2
      M3=M-3
      M4=M-4
      M5=M-5
      NNN=NN+1
      N=NN-1
      N1=N-1
      N2=N-2
      N3=N-3
      N4=N-4
      N5=N-5
      DO 1 K=2,5
      J=NNN-K
      YS(K)=10.*Y(7,J)-72.*Y(6,J)+225.*Y(5,J)-400.*Y(4,J)
     1+450.*Y(3,J)-360.*Y(2,J)
    1 CONTINUE
      X(1,NN)=0.
      X(1,N)=-(646.*YS(2)-264.*YS(3)+106.*YS(4)-19.*YS(5)
     1)/43200.
      X(1,N1)=-(124.*YS(2)+24.*YS(3)+4.*YS(4)-YS(5))/
     15400.
      X(1,N2)=-(306.*YS(2)+216.*YS(3)+126.*YS(4)-9.*YS(5)
     1)/14400.
      X(1,N3)=-(32.*(YS(2)+YS(4))+12.*YS(3)+7.*YS(5))/
     11350.
      DO 3 II=1,N4
      J=N3-II
      DO 2 K=1,4
      YS(K)=YS(K+1)
    2 CONTINUE
      YS(5)=10.*Y(7,J)-72.*Y(6,J)+225.*Y(5,J)-400.*Y(4,J)
     1+450.*Y(3,J)-360.*Y(2,J)
      X(1,J)=X(1,J+4)-(32.*(YS(2)+YS(4))+7.*(YS(1)+YS(5))
     1+12.*YS(3))/1350.
    3 CONTINUE
      DO 7 II=1,NN
      J=NNN-II
      JMM=J.EQ.NN
```

```
      JM=J.EQ.N
      JM1=J.EQ.N1
      JJG=J.GT.3.AND.J.LT.N1
      JJ3=J.EQ.3
      JJ2=J.EQ.2
      JJ1=J.EQ.1
      DO 4 K=2,5
      IF(JMM)YP(K)=10.*Y(K,N5)-72.*Y(K,N4)+225.*Y(K,N3)
     1-400.*Y(K,N2)+450.*Y(K,N1)-360.*Y(K,N)
      IF(JM)YP(K)=77.*Y(K,N)-150.*Y(K,N1)+100.*Y(K,N2)
     1-50.*Y(K,N3)+15.*Y(K,N4)-2.*Y(K,N5)
      IF(JM1)YP(K)=24.*Y(K,N)+35.*Y(K,N1)-80.*Y(K,N2)
     1+30.*Y(K,N3)-8.*Y(K,N4)+Y(K,N5)
      IF(JJG)YP(K)=Y(K,J+3)+9.*(Y(K,J-2)-Y(K,J+2))
     1+45.*(Y(K,J+1)-Y(K,J-1))-Y(K,J-3)
      IF(JJ3)YP(K)=-24.*Y(K,2)-35.*Y(K,3)+80.*Y(K,4)
     1-30.*Y(K,5)+8.*Y(K,6)-Y(K,7)
      IF(JJ2)YP(K)=-77.*Y(K,2)+150.*Y(K,3)-100.*Y(K,4)
     1+50.*Y(K,5)-15.*Y(K,6)+2.*Y(K,7)
      IF(JJ1)YP(K)=360.*Y(K,2)-450.*Y(K,3)+400.*Y(K,4)
     1-225.*Y(K,5)+72.*Y(K,6)-10.*Y(K,7)
    4 CONTINUE
      X(2,J)=X(1,J)+(646.*YP(2)-264.*YP(3)+106.*YP(4)
     1-19.*YP(5))/43200.
      X(3,J)=X(1,J)+(124.*YP(2)+24.*YP(3)+4.*YP(4)
     1-YP(5))/5400.
      X(4,J)=X(1,J)+(306.*YP(2)+216.*YP(3)+126.*YP(4)
     1-9.*YP(5))/14400.
      X(5,J)=X(1,J)+(32.*(YP(2)+YP(4))+12.*YP(3)+7.*YP(5)
     1)/1350.
      DO 6 K=6,M
      DO 5 I=1,4
      YP(I)=YP(I+1)
    5 CONTINUE
      IF(JMM)YP(5)=10.*Y(K,N5)-72.*Y(K,N4)+225.*Y(K,N3)
     1-400.*Y(K,N2)+450.*Y(K,N1)-360.*Y(K,N)
      IF(JM)YP(5)=77.*Y(K,N)-150.*Y(K,N1)+100.*Y(K,N2)
     1-50.*Y(K,N3)+15.*Y(K,N4)-2.*Y(K,N5)
      IF(JM1)YP(5)=24.*Y(K,N)+35.*Y(K,N1)-80.*Y(K,N2)
     1+30.*Y(K,N3)-3.*Y(K,N4)+Y(K,N5)
      IF(JJG)YP(5)=Y(K,J+3)+9.*(Y(K,J-2)-Y(K,J+2))
     1+45.*(Y(K,J+1)-Y(K,J-1))-Y(K,J-3)
      IF(JJ3)YP(5)=-24.*Y(K,2)-35.*Y(K,3)+80.*Y(K,4)
     1-30.*Y(K,5)+8.*Y(K,6)-Y(K,7)
      IF(JJ2)YP(5)=-77.*Y(K,2)+150.*Y(K,3)-100.*Y(K,4)
     1+50.*Y(K,5)-15.*Y(K,6)+2.*Y(K,7)
      IF(JJ1)YP(5)=360.*Y(K,2)-450.*Y(K,3)+400.*Y(K,4)
     1-225.*Y(K,5)+72.*Y(K,6)-10.*Y(K,7)
      X(K,J)=X(K-4,J)+(32.*(YP(2)+YP(4))+7.*(YP(1)
     1+YP(5))+12.*YP(3))/1350.
```

```
6 CONTINUE
7 CONTINUE
  RETURN
  END
```

```
      SUBROUTINE X5(NC,NR)
C
C     NUMERICAL INTEGRATION FOR THE X-COORDINATES
C     FOR THE CANAL PROBLEM.  THE Y-COORDINATES HAVE
C     BEEN COMPUTED USING THE FIVE-POINT FORMULA.
C
      DIMENSION X(11,41),Y(11,41)
      COMMON X,Y
      N=NC-1
      N1=N-1
      N2=N-2
      N3=N-3
C
C     X ALONG THE FIRST ROW.
C
      X(1,NC)=0.
      YSB=-48.*Y(2,N)+36.*Y(3,N)-16.*Y(4,N)+3.*Y(5,N)
      YSC=-48.*Y(2,N1)+36.*Y(3,N1)-16.*Y(4,N1)+3.*Y(5,N1)
      X(1,N)=(-8.*YSB+YSC)/144.
      X(1,N1)=-(4.*YSB+YSC)/36.
      DO 1 J=1,N2
      K=N1-J
      YSA=YSB
      YSB=YSC
      YSC=-48.*Y(2,K)+36.*Y(3,K)-16.*Y(4,K)+3.*Y(5,K)
      X(1,K)=X(1,K+2)-(YSA+4.*YSB+YSC)/36.
    1 CONTINUE
C
C     X ALONG THE (N+1)'ST COLUMN.
C
      YPB=-48.*Y(2,N)+36.*Y(2,N1)-16.*Y(2,N2)+3.*Y(2,N3)
      YPC=-48.*Y(3,N)+36.*Y(3,N1)-16.*Y(3,N2)+3.*Y(3,N3)
      X(2,NC)=(8.*YPB-YPC)/144.
      X(3,NC)=(4.*YPB+YPC)/36.
      DO 2 I=4,NR
      YPA=YPB
      YPB=YPC
      YPC=-48.*Y(I,N)+36.*Y(I,N1)-16.*Y(I,N2)+3.*Y(I,N3)
      X(I,NC)=X(I-2,NC)+(YPA+4.*YPB+YPC)/36.
    2 CONTINUE
C
C     X ALONG THE N'TH COLUMN.
C
      YPB=10.*Y(2,N)-18.*Y(2,N1)+6.*Y(2,N2)-Y(2,N3)
      YPC=10.*Y(3,N)-18.*Y(3,N1)+6.*Y(3,N2)-Y(3,N3)
      X(2,N)=X(1,N)+(8.*YPB-YPC)/144.
      X(3,N)=X(1,N)+(4.*YPB+YPC)/36.
      DO 3 I=4,NR
      YPA=YPB
      YPB=YPC
```

```
      YPC=10.*Y(I,N)-18.*Y(I,N1)+6.*Y(I,N2)-Y(I,N3)
      X(I,N)=X(I-2,N)+(YPA+4.*YPB+YPC)/36.
    3 CONTINUE
C
C     X ALONG COLUMNS N-1 TO 3.
C
      DO 5 L=1,N3
      J=N-L
      J0=J-2
      J1=J-1
      J2=J+1
      J3=J+2
      YPB=Y(2,J0)+8.*(Y(2,J2)-Y(2,J1))-Y(2,J3)
      YPC=Y(3,J0)+8.*(Y(3,J2)-Y(3,J1))-Y(3,J3)
      X(2,J)=X(1,J)+(8.*YPB-YPC)/144.
      X(3,J)=X(1,J)+(4.*YPB+YPC)/36.
      DO 4 I=4,M2
      YPA=YPB
      YPB=YPC
      YPC=Y(I,J0)+8.*(Y(I,J2)-Y(I,J1))-Y(I,J3)
      X(I,J)=X(I-2,J)+(YPA+4.*YPB+YPC)/36.
    4 CONTINUE
    5 CONTINUE
C
C     X ALONG THE 2'ND COLUMN.
C
      YPB=-10.*Y(2,2)+18.*Y(2,3)-6.*Y(2,4)+Y(2,5)
      YPC=-10.*Y(3,2)+18.*Y(3,3)-6.*Y(3,4)+Y(3,5)
      X(2,2)=X(1,2)+(8.*YPB-YPC)/144.
      X(3,2)=X(1,2)+(4.*YPB+YPC)/36.
      DO 6 I=4,M2
      YPA=YPB
      YPB=YPC
      YPC=-10*Y(I,2)+18.*Y(I,3)-6.*Y(I,4)+Y(I,5)
      X(I,2)=X(I-2,2)+(YPA+4.*YPB+YPC)/36.
    6 CONTINUE
C
C     X ALONG THE 1'ST COLUMN.
C
      YPB=48.*Y(2,2)-36.*Y(2,3)+16.*Y(2,4)-3.*Y(2,5)
      YPC=48.*Y(3,2)-36.*Y(3,3)+16.*Y(3,4)-3.*Y(3,5)
      X(2,1)=X(1,1)+(8.*YPB-YPC)/144.
      X(3,1)=X(1,1)+(4.*YPB+YPC)/36.
      DO 7 I=4,M2
      YPA=YPB
      YPB=YPC
      YPC=48.*Y(I,2)-36.*Y(I,3)+16.*Y(I,4)-3.*Y(I,5)
      X(I,1)=X(I-2,1)+(YPA+4.*YPB+YPC)/36.
    7 CONTINUE
      RETURN
```

END

B29890